Fixed-rank matrix factorizations and Riemannian low-rank optimization*

B. Mishra† G. Meyer† S. Bonnabel‡ R. Sepulchre†

September 4, 2012

Abstract

Motivated by the problem of learning a linear regression model whose parameter is a large fixed-rank non-symmetric matrix, we consider the optimization of a smooth cost function defined on the set of fixed-rank matrices. We adopt the geometric optimization framework of optimization on Riemannian matrix manifolds. We study the underlying geometries of several well-known fixed-rank matrix factorizations and then exploit the Riemannian geometry of the search space in the design of a class of gradient descent and trust-region algorithms. The proposed algorithms generalize our previous results on fixed-rank symmetric positive semidefinite matrices, apply to a broad range of applications, scale to high-dimensional problems and confer a geometric basis to recent contributions on the learning of fixed-rank non-symmetric matrices. We make connections with existing algorithms in the context of low-rank matrix completion and discuss relative usefulness of the proposed framework. Numerical experiments suggest that the proposed algorithms compete with the state-of-the-art and that manifold optimization offers an effective and versatile framework for the design of machine learning algorithms that learn a fixed-rank matrix.

1 Introduction

The problem of learning a low-rank matrix is a fundamental problem arising in many modern machine learning applications such as collaborative filtering [RS05], classification with multiple classes [AFSU07], learning on pairs [ABEV09], dimensionality reduction [CHH07], learning of low-rank distances [KSD09, MBS11b] and low-rank similarity measures [SWC10], multi-task learning [EMP05, MMBS11], just to name a few. Parallel to the development of

*This paper presents research results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors. Bamdev Mishra is a research fellow of the Belgian National Fund for Scientific Research (FNRS).

†Department of Electrical Engineering and Computer Science, University of Liège, 4000 Liège, Belgium (B.Mishra@ulg.ac.be, Gillesmy@gmail.com, R.Sepulchre@ulg.ac.be).

‡Robotics center Mines ParisTech Boulevard Saint-Michel, 60, 75272 Paris, France (Silvere.Bonnabel@mines-paristech.fr).

these new applications, the ever-growing size and number of large-scale datasets demands machine learning algorithms that can cope with large matrices. Scalability to high-dimensional problems is therefore a crucial issue in the design of algorithms that learn a low-rank matrix. Motivated by the above applications, the paper focuses on the following optimization problem

$$\min_{\mathbf{W}\in\mathcal{F}(r,d_1,d_2)} f(\mathbf{W}), \tag{1}$$

where $f : \mathbb{R}^{d_1\times d_2} \to \mathbb{R}$ is a smooth cost function and the search space is the set of fixed-rank non-symmetric real matrices,

$$\mathcal{F}(r, d_1, d_2) = \{\mathbf{W} \in \mathbb{R}^{d_1\times d_2} | \text{rank}(\mathbf{W}) = r\}.$$

We show in Section 2 that the considered optimization problem (1) encompasses various modern machine learning applications. We tackle problem (1) in a Riemannian framework, that is, by solving an unconstrained optimization on a Riemannian manifold in bijection with the nonlinear space $\mathcal{F}(r, d_1, d_2)$.

The paper follows and builds upon a number of recent contributions in that direction: the Ph.D. thesis [Mey11] and several papers by the authors [MBS11a, MBS11b, MMBS11, MMS11, Jou09]. The main contribution of this paper is to emphasize the common framework that underlines those contributions, with the aim of illustrating the versatile framework of Riemannian optimization for constrained optimization. Necessary ingredients to perform both first-order and second-order optimization are listed. An attempt is also made to classify the existing algorithms into various geometries and show the common structure that connects them all. Scalability to large dimension problems is shown in numerical examples. The generic performance of all algorithms is similar but strong gains can be obtained by adapting geometries and parameterizations to specific problems, in the same way as different families of first-order or second-order optimization algorithms adapt to specific problems.

2 Motivation and applications

In this section, a number of modern machine learning applications are cast as an optimization problem on the set of fixed-rank non-symmetric matrices.

2.1 Low-rank matrix completion

The problem of low-rank matrix completion amounts to estimating the missing entries of a matrix from a very limited number of its entries. There has been a large number of research contributions on this subject over the last few years, addressing the problem both from a theoretical [CR08, Gro11] and from algorithmic point of view [RS05, CCS08, LB09, MJD09, KMO10, SE10, JMD10, MHT10, BA11]. An important and popular application of the low-rank matrix completion problem is collaborative filtering [RS05, ABEV09].

Let $\mathbf{W}^\star \in \mathbb{R}^{d_1\times d_2}$ be a matrix whose entries $\mathbf{W}^\star_{ij}$ are only given for some indices $(i, j) \in \Omega$, where Ω is a subset of the complete set of indices $\{(i, j) : i \in \{1, ..., d_1\} \text{ and } j \in \{1, ..., d_2\}\}$.

Low-rank matrix completion amounts to solving the following optimization problem

$$\begin{array}{ll} \min_{\mathbf{W}\in\mathbb{R}^{d_1\times d_2}} & \frac{1}{|\Omega|}\|\mathcal{P}_\Omega(\mathbf{W}) - \mathcal{P}_\Omega(\mathbf{W}^\star)\|_F^2 \\ \text{subject to} & \text{rank}(\mathbf{W}) = r, \end{array} \tag{2}$$

where the function $\mathcal{P}_\Omega(\mathbf{W}_{ij}) = \mathbf{W}_{ij}$ if $(i,j) \in \Omega$ and $\mathcal{P}_\Omega(\mathbf{W}_{ij}) = 0$ otherwise and the norm $\|\cdot\|_F$ is *Frobenius* norm. The function $\mathcal{P}_\Omega$ is also called the *orthogonal sampling operator* and $|\Omega|$ is the cardinality of the set Ω (equal to the number of known entries).

The rank constraint captures redundant patterns in $\mathbf{W}^\star$ and ties the known and unknown entries together. The number of given entries $|\Omega|$ is typically much smaller than $d_1 d_2$, the total number of entries in $\mathbf{W}^*$. Recent contributions provide conditions on $|\Omega|$ under which exact reconstruction is possible from entries sampled uniformly and at random [CR08, CCS08, KMO10].

2.2 Learning on data pairs

Given two types of data $\mathbf{x} \in \mathcal{X}$ (say $\mathbb{R}^{d_1}$) and $\mathbf{z} \in \mathcal{Z}$ (say $\mathbb{R}^{d_2}$) associated with two types of samples and also given are the associated scalar observations $y \in \mathbb{R}$, learning on data pairs amounts to learning a predictive model $\hat{y} : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ from training examples $\{(\mathbf{x}_i, \mathbf{z}_i, y_i)\}_{i=1}^n$. When the predictive model $\hat{y}$ is chosen as the bilinear form

$$\hat{y} = \mathbf{x}^T \mathbf{W} \mathbf{z} \quad \text{with} \quad \mathbf{W} \in \mathcal{F}(r, d_1, d_2),$$

then the problem boils down to the optimization problem (1)

$$\min_{\mathbf{W}\in\mathcal{F}(r,d_1,d_2)} \sum_{i=1}^n \{\ell(\hat{y}_i, y_i)\} \tag{3}$$

where the loss function $\ell(\hat{y}, y)$ penalizes the discrepancy between an observation y and the value that is predicted by the model $\hat{y}$. An application of this setup is the inference of edges in bipartite or directed graphs. Such a problem arises in bioinformatics for the identification of interactions between drugs and target proteins, micro-RNA and genes or genes and diseases [YAG+08, BY09]. Another application is concerned with image domain adaptation [KSD11] where a transformation $\mathbf{x}^T\mathbf{W}\mathbf{z}$ is learned between labeled images $\mathbf{x}$ from a source domain $\mathcal{X}$ and labeled images $\mathbf{z}$ from a target domain $\mathcal{Z}$. The transformation $\mathbf{W}$ is used on any new input data to map the data from one domain to the other. A potential interest of the rank constraint in these applications is to address problems with a high-dimensional feature space and perform dimensionality reduction on the two data domains.

2.3 Multivariate linear regression

Given matrices $\mathbf{Y} \in \mathbb{R}^{n\times k}$ (output space) and $\mathbf{X} \in \mathbb{R}^{n\times q}$ (input space), we seek to learn a weight/coefficient matrix $\mathbf{W} \in \mathbb{R}^{q\times k}$ that minimizes the discrepancy between $\mathbf{Y}$ and $\mathbf{XW}$ [YELM07]. Here n is the number of observations, q is the number of predictors and k is the number of responses. One popular approach to multivariate linear regression problem is by minimizing a *quadratic loss* function. Note that in various applications *responses* are related

and may therefore, be represented with much fewer coefficients. From an optimization point to view this corresponds to finding a low-rank coefficient matrix. The papers [YELM07, AFSU07], thus, motivate the rank-constrained optimization problem formulation. In the present framework it is defined as,

$$\min_{\mathbf{W}\in\mathcal{F}(r,q,k)} \quad \|\mathbf{Y}-\mathbf{X}\mathbf{W}\|_F^2.$$

Though the quadratic loss function is shown here, the optimization setup extends to other smooth loss functions as well.

2.4 Numerical comparisons: Matrix completion as a benchmark

To illustrate the notions presented in this paper, we consider the problem of low-rank matrix completion as the benchmark application. The objective function is a smooth least square function and the search space is the space of fixed-rank matrices. It is an optimization problem that has attracted a lot of attention in recent years.

All simulations are performed in MATLAB on a 2.53 GHz Intel Core i5 machine with 4 GB of RAM. We use MATLAB codes of the benchmark algorithms for our numerical studies. For each example, a $d_1 \times d_2$ random matrix of rank r is generated according to a Gaussian distribution with zero mean and unit standard deviation and a fraction of the entries are randomly removed with uniform probability. The dimensions of $d_1 \times d_2$ matrices of rank r is $(d_1+d_2-r)r$. The over-sampling (OS) ratio determines the number of entries that are known. A OS $= 6$ means that $6(d_1 + d_2 - r)r$ number of randomly and uniformly selected entries are known a priori out of $d_1 d_2$ entries. Matrix reconstruction results have been proved in the context of fixed-rank optimization [KMO10] with an appropriate initialization. All algorithms (including competing methods) are initialized as proposed in [KMO10]. Numerical codes for the proposed algorithms are available from the first author's homepage.

We consider the problem of completing a 32000×32000 matrix $\mathbf{W}^\star$ of rank 5 as the running example in many comparisons. The over-sampling ratio OS is 8 implying that 0.25% (2559800) of entries are randomly and uniformly revealed. The maximum number of iterations is fixed at 500 and the tolerance for the objective function is set at $\frac{1}{|\Omega|}\|\mathcal{P}_\Omega(\mathbf{W})-\mathcal{P}_\Omega(\mathbf{W}^\star)\|_F^2 \leq 10^{-20}$ (10^{-25} for trust-region algorithms). A high tolerance is needed to observe the asymptotic rate of convergence of algorithms. For numerical comparisons with various competing algorithms we use the MATLAB codes supplied on authors' homepages.

3 From matrix factorization to Riemannian quotient geometry

We review three matrix factorizations for fixed-rank non-symmetric matrices and study the geometry of the resulting search space. Factorization models lead to product spaces of well known matrix manifolds. The quotient nature of the search space stems from the fact that a given element $\mathbf{W} \in \mathcal{F}(r, d_1, d_2)$ is represented by an entire equivalence class of matrices due to intrinsic invariance properties of the factorization (Figure 1). Understanding invariance is critical in designing algorithms [AMS08, AILH09]. An important observation is that the local

minima are not isolated in the product space of factorization models. We seek to optimize the cost function in an abstract search space where the local minima are isolated. Such a space arises naturally in the factorization schemes as shown below and with a proper metric has the structure of a Riemannian manifold [AMS08].

The fixed-rank matrix factorizations of interest are rooted in the thin singular value decomposition of a r-rank matrix $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{U}$ is a d_1-by-r matrix with orthogonal columns, that is, an element of the Stiefel manifold $\mathrm{St}(r, d_1) = \{\mathbf{U} \in \mathbb{R}^{d_1 \times r} : \mathbf{U}^T\mathbf{U} = \mathbf{I}\}$, $\mathbf{\Sigma} \in \mathrm{Diag}_+(r)$ is a r-by-r diagonal matrix with positive entries and $\mathbf{V} \in \mathrm{St}(r, d_2)$. The singular value decomposition (SVD) exists for any matrix $\mathbf{W} \in \mathcal{F}(r, d_1, d_2)$ [GVL96].

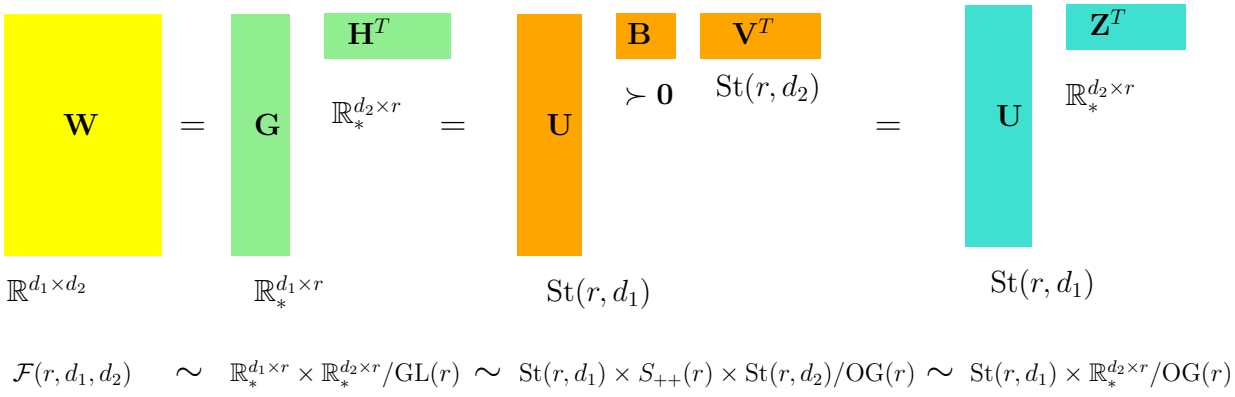

Figure 1: The considered fixed-rank factorizations admit a quotient manifold geometry due to intrinsic invariance properties of factorization models.

3.1 Full-rank factorization (beyond Cholesky-type decomposition)

The first factorization is obtained when the singular value decomposition (SVD) is rearranged as

$$\mathbf{W} = (\mathbf{U}\mathbf{\Sigma}^{\frac{1}{2}})(\mathbf{\Sigma}^{\frac{1}{2}}\mathbf{V}^T) = \mathbf{G}\mathbf{H}^T,$$

where $\mathbf{G} = \mathbf{U}\mathbf{\Sigma}^{\frac{1}{2}} \in \mathbb{R}_*^{d_1 \times r}$, $\mathbf{H} = \mathbf{V}\mathbf{\Sigma}^{\frac{1}{2}} \in \mathbb{R}_*^{d_2 \times r}$ and $\mathbb{R}_*^{d \times r}$ is the set of full column rank $d \times r$ matrices, also known as *full-rank matrix factorization*. The resulting factorization is not unique because the transformation

$$(\mathbf{G}, \mathbf{H}) \mapsto (\mathbf{G}\mathbf{M}^{-1}, \mathbf{H}\mathbf{M}^T), \tag{4}$$

where $\mathbf{M} \in \mathrm{GL}(r) = \{\mathbf{M} \in \mathbb{R}^{r \times r} : \det(\mathbf{M}) \neq 0\}$ leaves the original matrix $\mathbf{W}$ unchanged [PO99]. The classical remedy to remove this indeterminacy is Cholesky factorization, which requires further (triangular-like) structure in the factors. LU decomposition is a way forward [GVL96]. In contrast, we encode the invariance map (4) in an abstract search space by optimizing over a set of equivalence classes defined as

$$[\mathbf{W}] = [(\mathbf{G}, \mathbf{H})] = \{(\mathbf{G}\mathbf{M}^{-1}, \mathbf{H}\mathbf{M}^T) : \mathbf{M} \in \mathrm{GL}(r)\}. \tag{5}$$

The set of equivalence classes is termed as the quotient space of $\overline{\mathcal{W}}$ by $\mathrm{GL}(r)$ and is denoted as

$$\mathcal{W} := \overline{\mathcal{W}}/\mathrm{GL}(r), \tag{6}$$

where the total space $\overline{\mathcal{W}}$ is the product space $\mathbb{R}_*^{d_1 \times r} \times \mathbb{R}_*^{d_2 \times r}$.

Among the set of equivalence$[(\mathbf{G}, \mathbf{H})]$, a numerically well-conditioned representative is obtained by choosing $(\mathbf{G}, \mathbf{H})$ such that $\|\mathbf{G}\|_F = \|\mathbf{H}\|_F$. This *balancing update* condition encodes the fact that both factors $\mathbf{G}$ and $\mathbf{H}$ have a comparable weight in the factorization. The usefulness of using the quotient geometry with the numerically-conditioned factors is demonstrated in Section 5.2.

3.2 Polar factorization (beyond SVD)

The second quotient structure for the set $\mathcal{F}(r, d_1, d_2)$ is obtained by considering the following group action on the SVD [BS09],

$$(\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}) \mapsto (\mathbf{UO}, \mathbf{O}^T\mathbf{\Sigma O}, \mathbf{VO}),$$

where $\mathbf{O}$ is any $r \times r$ orthogonal matrix, that is, any element of the orthogonal group

$$\mathcal{O}(r) = \{\mathbf{O} \in \mathbb{R}^{r \times r} : \mathbf{O}^T\mathbf{O} = \mathbf{OO}^T = \mathbf{I}\}.$$

This results in *polar factorization*

$$\mathbf{W} = \mathbf{UBV}^T,$$

where $\mathbf{B}$ is now a $r \times r$ symmetric positive definite matrix, that is, an element of

$$S_{++}(r) = \{\mathbf{B} \in \mathbb{R}^{r \times r} : \mathbf{B}^T = \mathbf{B} \succ 0\}. \tag{7}$$

The polar factorization is close to the original purpose of singular value decomposition as identifying the geometries invariants of linear transformations [GVL96]. The choice of $\mathbf{B}$ being positive definite rather than $\mathbf{\Sigma}$ diagonal gives more flexibility in the optimization and removes the discrete symmetries induced by interchanging the order on the singular values. Empirical evidence to support the choice of $S_{++}(r)$ over $\mathrm{Diag}_+(r)$ (set of diagonal matrices with positive entries) for the middle factor $\mathbf{B}$ is presented in Section 5.3.

The product space is $\mathrm{St}(r, d_1) \times S_{++}(r) \times \mathrm{St}(r, d_2)$. The set $\mathcal{F}(r, d_1, d_2)$ is the quotient $\overline{\mathcal{W}}/\mathcal{O}(r)$, that is, the set of equivalence classes defined as

$$[\mathbf{W}] = [(\mathbf{U}, \mathbf{B}, \mathbf{V})] = \{(\mathbf{UO}, \mathbf{O}^T\mathbf{BO}, \mathbf{VO}) : \ \mathbf{O} \in \mathcal{O}(r)\}. \tag{8}$$

A relevant property of the factorization is that regularizing the norm of $\mathbf{W}$ is very cheap because it reduces to regularizing the norm of $\mathbf{B}$ which is of size $r \ll \min(d_1, d_2)$. For example, adding a regularization term proportional to the Frobenius norm $\|\mathbf{W}\|_F$ to the cost function is equivalent to adding a regularization term proportional to $\|\mathbf{B}\|_F$. Likewise, a regularization on the nuclear norm $\|\mathbf{W}\|_* = \sum \sigma_i$ where $\sigma_i s$ are the singular values of $\mathbf{W}$, is equivalent to a regularization on the trace of $\mathbf{B}$ i.e., $\|\mathbf{UBV}^T\|_* = \mathrm{Tr}(\mathbf{B})$ [MMBS11].

3.3 Subspace-projection factorization (beyond QR decomposition)

The third low-rank factorization is obtained from the SVD when two factors are grouped together,

$$\mathbf{W} = \mathbf{U}(\mathbf{\Sigma}\mathbf{V}^T) = \mathbf{U}\mathbf{Z}^T,$$

where $\mathbf{U} \in \mathrm{St}(r, d_1)$ and $\mathbf{Z} \in \mathbb{R}_*^{d_2 \times r}$. The column subspace of $\mathbf{W}$ matrix is represented by $\mathbf{U}$ while $\mathbf{Z}$ is the (left) *projection* or *coefficient* matrix of $\mathbf{W}$. The factorization is not unique as it is invariant with respect to the group action $(\mathbf{U}, \mathbf{Z}) \mapsto (\mathbf{UO}, \mathbf{ZO})$, whenever $\mathbf{O} \in \mathcal{O}(r)$. The classical remedy to remove this indeterminacy is the QR factorization for which $\mathbf{Z}$ is chosen upper triangular [GVL96]. We work with the set of equivalence classes

$$[\mathbf{W}] = [(\mathbf{U}, \mathbf{Z})] = \{(\mathbf{UO}, \mathbf{ZO}) : \ \mathbf{O} \in \mathcal{O}(r)\} \tag{9}$$

in the total space $\overline{\mathcal{W}} := \mathrm{St}(r, d_1) \times \mathbb{R}^{d_2 \times r}$, viewing the search space as the quotient space

$$\mathcal{W} = \overline{\mathcal{W}}/\mathcal{O}(r). \tag{10}$$

Recent contributions using this factorization include [BA11, SE10]. See the discussion in Section 5.1 for a comparative review.

4 Manifold-based optimization

Classical optimization algorithms such as penalty methods, barrier methods or augmented Lagrangian methods [NW06] generally deal with structured matrix search spaces by means of explicit constraints or penalty terms expressed as a function of the decision variable. These methods turn a constrained optimization problem into a sequence of unconstrained optimization problems for which classical unconstrained optimization techniques are applied. An alternative is to embed the constraints into the search space and to solve an unconstrained optimization problem on the constrained search space. This approach is taken by optimization algorithms on matrix manifolds [AMS08].

In a nutshell, a manifold $\mathcal{W}$ is a set endowed with a compatible differentiable structure. Once a manifold is endowed with a compatible differentiable structure, calculations can be performed and the classical geometric objects of optimization such as derivatives, gradients or Hessians all admit a generalization to manifolds.

Important class of manifolds include embedded submanifolds and quotient manifolds. Embedded submanifolds can be viewed as a generalization of the notion of surface in the Euclidean space. They are defined by means of an explicit set of algebraic constraints in matrix space. This general concept applies straight to the Stiefel manifold $\mathrm{St}(r, d)$ that is regarded as a submanifold embedded in the Euclidean space $\mathbb{R}^{d \times r}$. Another example is the $d-1$-dimensional unit sphere $\mathcal{S}^{d-1}$ embedded in $\mathbb{R}^d$, which coincides with $\mathrm{St}(1, d)$. Likewise, the set of d-by-d orthogonal matrices $\mathcal{O}(d)$ can be regarded as an embedded submanifold of $\mathbb{R}^{d \times d}$ and coincides with $\mathrm{St}(d, d)$. The set, $\mathcal{F}(r, d_1, d_2)$, of $r-$ rank $\mathbb{R}^{d_1 \times d_2}$ matrices also forms a smooth manifold embedded in the Euclidean space $\mathbb{R}^{d_1 \times d_2}$ and therefore, can also be

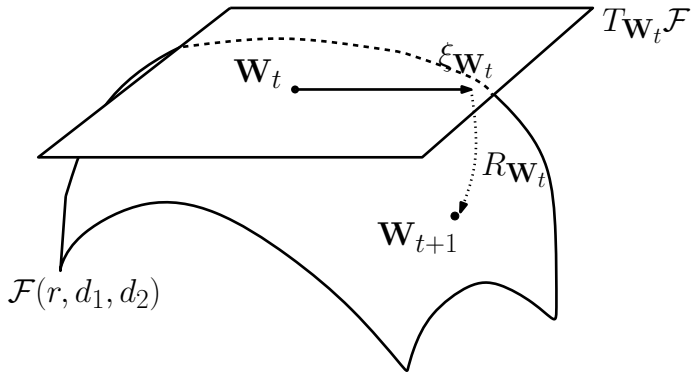

Figure 2: A line-search on a Riemannian embedded submanifold $\mathcal{F}(r, d_1, d_2)$ in the Euclidean space $\mathbb{R}^{d_1 \times d_2}$. The search direction $\xi_{\mathbf{W}_t}$ belongs to the tangent space $T_{\mathbf{W}_t}\mathcal{F}$. The retraction mapping R pulls back the new iterate onto the manifold $\mathcal{F}(r, d_1, d_2)$.

viewed as an *embedded submanifold* [SWC10, Van11]. See Figure 2. Details in Section 5.4. This geometry is different from the quotient geometry proposed in Section 3.

The present paper focuses on the quotient manifold geometries induced by the factorization models of Section 3. Each point of a (matrix) quotient manifold represents an entire equivalence class of matrices. Abstract geometric objects on the quotient manifold can be defined by means of matrix representatives, provided that their definitions do not depend on a particular choice for the representative within an equivalence class.

Let $\overline{\mathcal{W}}$ be the total space that is equipped with an equivalence relation $\sim$. For example, with the subspace-projection factorization model, each point $\bar{x} \in \overline{\mathcal{W}}$ has the matrix representation $(\mathbf{U}, \mathbf{Z})$ such that $\mathbf{W} = \mathbf{U}\mathbf{Z}^T$. The equivalence class (or *fiber*) of a given point $\bar{x} \in \overline{\mathcal{W}}$ is defined by the set

$$= \{\overline{y} \in \overline{\mathcal{W}} : \bar{y} \sim \bar{x}\}$$

where $x \in \mathcal{W}$ is the equivalence class $[\bar{x}]$. By extension, the set

$$\mathcal{W} := \overline{\mathcal{W}}/\sim \triangleq \{[\mathbf{W}] : \mathbf{W} \in \overline{\mathcal{W}}\},$$

is the quotient manifold of $\overline{\mathcal{W}}$ by $\sim$. The mapping $\pi : \overline{\mathcal{W}} \to \mathcal{W}$ is called the quotient map or canonical projection. Clearly, we have $\pi(\bar{x}) = \pi(\bar{y})$ if and only if $\bar{x} \sim \bar{y}$. The set $\overline{\mathcal{W}}$ is the *total space* of the quotient manifold $\mathcal{W}$.

A popular example of quotient manifold is the Grassmann manifold $\mathrm{Gr}(r, d)$, that is, the set of r-dimensional subspaces in $\mathbb{R}^{d \times r}$. $\mathrm{Gr}(r, d) \approx \mathbb{R}_*^{d \times r}/\mathcal{O}(r)$ where $\mathbb{R}_*^{d \times r}$ is the set of full column rank $d \times r$ matrices and $\mathcal{O}(r)$ is the set of $r \times r$ orthogonal matrices. Indeed, each subspace can be defined by a r-dimensional orthogonal frame up to a rotation matrix $\mathbf{O} \in \mathcal{O}(r)$. The reader is referred to the papers of [EAS98] or [AMS04] for alternative characterizations of the Grassmann manifold $\mathrm{Gr}(r, d)$ as a quotient manifold.

Notions on quotient manifold

For quotient manifolds $\mathcal{W} = \overline{\mathcal{W}}/\sim$ a tangent vector $\xi_x \in T_x\mathcal{W}$ at $x = [\bar{x}]$ is restricted to the directions that do not induce a displacement along the set of equivalence classes. This

is achieved by decomposing the tangent space in the total space $T_{\bar{x}}\overline{\mathcal{W}}$ into complementary spaces

$$T_{\mathbf{W}}\overline{\mathcal{W}} = \mathcal{V}_{\mathbf{W}}\overline{\mathcal{W}} \oplus \mathcal{H}_{\mathbf{W}}\overline{\mathcal{W}}.$$

Refer Figure 3 for a graphical illustration. The *vertical space* $\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}$ is the set of directions

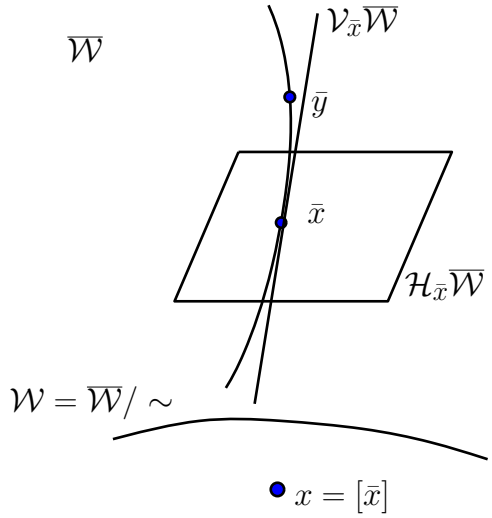

Figure 3: The quotient space. The points $\bar{y}$ and $\bar{x}$ in $\overline{\mathcal{W}}$ belonging to the same equivalence class are represented by a single point $[x]$ in the quotient space $\mathcal{W}$.

that contains tangent vectors to the equivalence classes. The *horizontal space* $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ is the complement of $\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}$ in $T_{\bar{x}}\overline{\mathcal{W}}$. The horizontal space provides a representation of the abstract tangent vectors to the quotient space, i.e., $T_x\mathcal{W}$. Indeed, displacements in the vertical space leave the matrix $\mathbf{W}$ (matrix representation of the point $\bar{x}$) unchanged, which suggests to restrict both tangent vectors and metric to the horizontal space. Once $T_{\bar{x}}\overline{\mathcal{W}}$ is endowed with a horizontal distribution $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$, a given tangent vector $\xi_x \in T_x\mathcal{W}$ at x in the quotient manifold $\mathcal{W}$ is uniquely represented by a tangent vector $\bar{\xi}_{\bar{x}} \in \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ in the total space $\overline{\mathcal{W}}$.

The tangent vector $\bar{\xi}_{\bar{x}} \in \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ is called the *horizontal lift* of ξ_x at $\bar{x}$. Provided that the metric defined in the total space is invariant along the set of equivalence classes. A metric $\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\zeta}_{\bar{x}})$ in the total space defines a metric g_x on the quotient manifold. Namely,

$$g_x(\xi_x, \zeta_x) := \bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\zeta}_{\bar{x}}) \tag{11}$$

where ξ_x and ζ_x are the tangent vectors in $T_x\mathcal{W}$ and $\xi_{\bar{x}}$ and $\zeta_{\bar{x}}$ are their horizontal lifts in $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$.

Natural displacements at $\bar{x}$ in a direction $\bar{\xi}_{\bar{x}} \in \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ on the manifold are performed by following geodesics (paths of shortest length on the manifold) starting from $\bar{x}$ and tangent to $\xi_{\bar{x}}$. This is performed by means of the exponential map

$$\bar{x}_{t+1} = \mathrm{Exp}_{\bar{x}_t}(s_t \bar{\xi}_{\bar{x}_t}),$$

which induces a line-search algorithm along geodesics with the step size s_t. However, the geodesics are generally expensive to compute and, in many cases, are not available in closed-form. A more general update formula is obtained if we relax the constraint of moving along

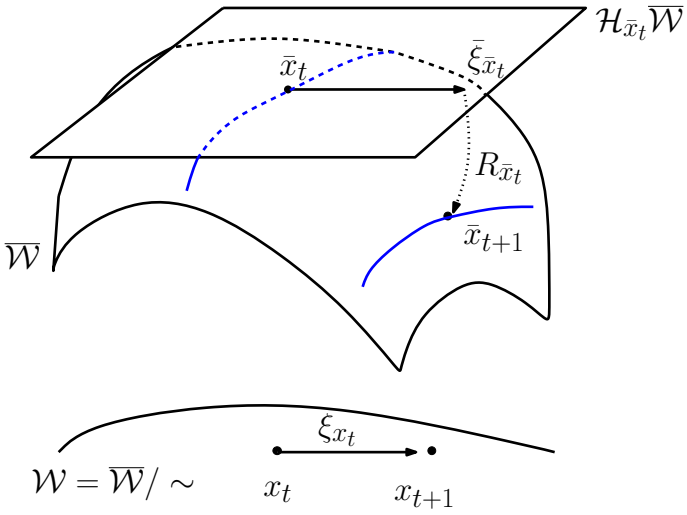

Figure 4: A line-search on a Riemannian quotient manifold $\mathcal{W}$ in the total space $\overline{\mathcal{W}}$. Conceptually, we move on the quotient manifold from x_t to x_{t+1} but computationally in the total space $\overline{\mathcal{W}}$. The search direction is $\bar{\xi}_{\bar{x}_t}$ and belongs to the horizontal space $\mathcal{H}_{\bar{x}_t}\overline{\mathcal{W}}$ at point $\bar{x}_t$. The retraction mapping maintains feasibility of the iterates. The blue line denotes the equivalence class $[\bar{x}_t]$.

geodesics. The retraction mapping $R_{\bar{x}_t}(s_t\xi_{\bar{x}_t})$ at $\bar{x}_t$ locally approximates the exponential mapping [AMS08]. It provides a numerically attractive alternative to the exponential mapping in the design of optimization algorithms on manifolds, as it reduces the computational complexity of the update while retaining the essential properties that ensure convergence results. A generic abstract line-search algorithm is, thus, based on the update formula

$$\bar{x}_{t+1} = R_{\bar{x}_t}(s_t\bar{\xi}_{\bar{x}_t}) \tag{12}$$

where s_t is the step-size. A good step-size is computed using the Armijo rule [NW06]. We use the information of the previous step-size to make the initial guess by using the following simple *adaptive step-size update* procedure.

$$\begin{aligned}
\text{Given}: \quad & \hat{s}_t(\text{initial step}-\text{size guess}) \\
& j_t \ (\text{number of line}-\text{search}), \ \text{and} \\
& s_t(\text{optimal step}-\text{size}) \ \text{at iterate} \ t \\
\text{Then}: \quad & \text{initial step}-\text{size guess at} \ t+1 \ \text{iterate} \\
& \hat{s}_{t+1} = \begin{cases} 2\hat{s}_t, & j_t = 0 \\ 2s_t, & j_t \geq 2. \end{cases}
\end{aligned} \tag{13}$$

It ensures that on an average we keep the number of line-searches close to 1.

Similarly, second-order algorithms on the quotient manifold $\mathcal{W}$ are horizontally lifted and solved in the total space $\overline{\mathcal{W}}$. Additionally, we need to be able to compute the directional derivative of gradient along a search direction. This relationship is captured by *an affine connection* ∇ on the manifold. The vector field $\nabla_\eta\xi$ implies the *covariant derivative* of vector field η with respect to the vector field ξ. In the case of $\mathcal{W}$ being a Euclidean space, the affine

connection is standard

$$(\nabla_\xi \eta)_x = \lim_{t \to 0} \frac{\eta_{x+t\xi_x} - \eta_x}{t}.$$

However, for an arbitrary manifold there exists infinitely many different affine connections except for a specific connection called the *Levi-Civita* or *Riemannian* connection which is always unique. The properties of affine connections and Riemannian connection are in section 5.2 and Theorem 5.3.1 of [AMS08]. The Riemannian connection on the quotient manifold $\mathcal{W}$ is given in terms of the connection in the total space $\overline{\mathcal{W}}$ once the quotient manifold has the structure of a *Riemannian submersion* [AMS08].

Analogous to trust-region algorithms in the Euclidean space, trust-region algorithms on a quotient manifold with guaranteed quadratic rate convergence have been proposed in [ABG07, AMS08]. The trust-region subproblem on $\mathcal{W}$ is formulated as

$$\begin{array}{ll} \min_{\xi \in T_x\mathcal{W}} & \phi(x) + g_x(\xi, \mathrm{grad}\phi(x)) + \frac{1}{2} g_x(\xi, \mathrm{Hess}\phi(x)[\xi]) \\ \text{subject to} & g_x(\xi, \xi) \leq \delta. \end{array}$$

where δ is the trust-region radius and gradϕ and Hessϕ are the Riemannian gradient and Hessian on $\mathcal{W}$. The problem is horizontally lifted to the horizontal space $\mathcal{H}_{\bar{x}}\mathcal{W}$ [ABG07, BAG07]. Solving the above trust-region subproblem leads to a direction $\bar{\xi}$ that minimizes the quadratic model. The trust-region subproblem is solved efficiently by the generic solver GenRTR [BAG07]. The potential iterate on the manifold $\mathcal{W}$ is obtained using the retraction operator R. Depending on whether the decrease of the cost function sufficient or not, the potential iterate is accepted or rejected.

Numerical complexity

The numerical complexity per iteration of the proposed gradient descent and trust-region algorithms depends on the computational cost of various components described before. All manifold related operations are of linear complexity in d_1 and d_2. Other operations depend on the problem at hand and are computed in the search space $\overline{\mathcal{W}}$. With $r \ll \min\{d_1, d_2\}$ the computational burden on algorithms is considerably low [MMBS11, Mey11]. A tentative list of computational cost of geometry related operations is given in Section 5.1.

5 Riemannian geometries of rank-constrained matrices

In this section, we present the geometric concepts that allow for a systematic derivation of line-search and trust-region algorithms on each of the factorization models described in Section 3. The tools are generic in nature and do not depend on the cost function at hand. The illustrations, however, are shown on the low-rank matrix completion problem (see Section 2.1). The main concepts and notations have been introduced in Section 4. This section discusses the derivation of these objects. The developments are shown in detail for the subspace-projection factorization (Section 3.3). For the other two factorization models, full-rank factorization and polar factorization, the developments are similar and we present only the final expressions. Details can be found in [MMBS11, Mey11].

5.1 Geometry of algorithms using a subspace-projection factorization $\mathbf{W} = \mathbf{U}\mathbf{Z}^T$

We a seek a local minimum to the following optimization problem

$$\begin{array}{ll} \min_{\mathbf{U},\mathbf{Z}} & \bar{\phi}(\mathbf{U},\mathbf{Z}) \\ \text{subject to} & (\mathbf{U},\mathbf{Z}) \in \overline{\mathcal{W}} \end{array} \tag{14}$$

where $\overline{\mathcal{W}}$ is the total space $\mathrm{St}(r,d_1) \times \mathbb{R}^{d_2 \times r}$ and $\bar{\phi} : \overline{\mathcal{W}} \to \mathbb{R}$ is a smooth function. As mentioned in Section 3.3, the local minima of the above optimization problem are not isolated in the space $\overline{\mathcal{W}}$ and hence, we seek to solve the problem in the quotient space $\mathcal{W} = \overline{\mathcal{W}}/\mathcal{O}(r)$. The problem (14) is thus conceptually an *unconstrained* optimization problem on the quotient manifold $\mathcal{W}$ in which the minima are isolated. Computations are performed in the total space $\overline{\mathcal{W}}$, which is the product space of well-studied manifolds.

Tangent space of $\mathcal{W}$

Tangent vectors at a point $x \in \mathcal{W}$ have a matrix representation in the tangent space of the total space $\overline{\mathcal{W}}$. Because the total space is a product space $\mathrm{St}(r,d_1) \times \mathbb{R}_*^{d_2 \times r}$, its tangent space admits the decomposition at a point $\bar{x} = (\mathbf{U},\mathbf{Z})$

$$T_{\bar{x}}\overline{\mathcal{W}} = T_{\mathbf{U}}\mathrm{St}(r,d_1) \times T_{\mathbf{Z}}\mathbb{R}_*^{d_2 \times r}$$

and the following characterizations are well-known [EAS98, AMS08]. Note that $\mathbb{R}_*^{d_2 \times r}$ is an open subset of $\mathbb{R}^{d_2 \times r}$.

$$\begin{array}{rcl} T_{\mathbf{U}}\mathrm{St}(r,d_1) & = & \{\mathbf{U}\boldsymbol{\Omega} + \mathbf{U}_\perp\mathbf{K} | \boldsymbol{\Omega} \in S_{skew}(r), \mathbf{K} \in \mathbb{R}^{(d_1-r)\times r}\} \\ & = & \{\mathbf{Z}_{\mathbf{U}} - \mathbf{U}\mathrm{Sym}(\mathbf{U}^T\mathbf{Z}_{\mathbf{U}}) | \mathbf{Z}_{\mathbf{U}} \in \mathbb{R}^{n\times p}\} \\ T_{\mathbf{Z}}\mathbb{R}_*^{d_2\times r} & = & \mathbb{R}^{d_2\times r} \end{array}$$

where $S_{skew}(r)$ is the set of $r \times r$ skew-symmetric matrices, the operator $\mathrm{Sym}(\mathbf{A}) = \frac{\mathbf{A}+\mathbf{A}^T}{2}$ and $\mathbf{U}_\perp$ is the orthogonal complement of the space spanned by $\mathbf{U}$. Note that an arbitrary matrix $(\mathbf{Z}_{\mathbf{U}}, \mathbf{Z}_{\mathbf{Z}}) \in \mathbb{R}^{d_1\times r} \times \mathbb{R}^{d_2 \times r}$ is projected on the tangent space $T_{\bar{x}}\overline{\mathcal{W}}$ by the linear operation

$$\Psi_{\bar{x}}(\mathbf{Z}_{\mathbf{U}}, \mathbf{Z}_{\mathbf{Z}}) = (\mathbf{Z}_{\mathbf{U}} - \mathbf{U}\mathrm{Sym}(\mathbf{U}^T\mathbf{Z}_{\mathbf{U}}), \mathbf{Z}_{\mathbf{Z}}). \tag{15}$$

A matrix representation of the tangent space at the equivalence class $x = [\bar{x}] \in \mathcal{W}$ relies on the decomposition of $T_{\bar{x}}\overline{\mathcal{W}}$ into its *vertical* and *horizontal* subspaces. The vertical space $\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}$ is the subspace of $T_{\bar{x}}\overline{\mathcal{W}}$ that is tangent to the equivalence class $[\bar{x}]$

$$\mathcal{V}_{\bar{x}}\overline{\mathcal{W}} = \{(\mathbf{U}\boldsymbol{\Omega}, \mathbf{Z}\boldsymbol{\Omega}) | \boldsymbol{\Omega} \in S_{skew}(r)\}. \tag{16}$$

The horizontal space $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ is chosen such that

$$T_{\bar{x}}\overline{\mathcal{W}} = \mathcal{H}_{\bar{x}}\overline{\mathcal{W}} \oplus \mathcal{V}_{\bar{x}}\overline{\mathcal{W}}. \tag{17}$$

We choose $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ as the orthogonal complement of $\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}$ for the metric

$$\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}}) \quad = \quad \mathrm{Tr}(\bar{\xi}_{\mathbf{U}}^T\bar{\eta}_{\mathbf{U}}) + \mathrm{Tr}((\mathbf{Z}^T\mathbf{Z})^{-1}\bar{\xi}_{\mathbf{Z}}^T\bar{\eta}_{\mathbf{B}}), \tag{18}$$

which picks the normal metric of the Stiefel manifold [EAS98] and the *right-invariant* metric of $\mathbb{R}_*^{d_2\times r}$ [AMS04]. $\bar{\xi}_{\bar{x}}$ and $\bar{\eta}_{\bar{x}}$ are elements of $T_{\bar{x}}\overline{\mathcal{W}}$. The choice of this metric is motivated later in this section. With this choice, a horizontal tangent vector $\zeta_{\bar{x}}$ is any tangent vector $(\zeta_{\mathbf{U}}, \zeta_{\mathbf{Z}})$ belonging to the set

$$\mathcal{H}_{\bar{x}}\overline{\mathcal{W}} \quad = \quad \{(\zeta_{\mathbf{U}}, \zeta_{\mathbf{Z}}) \in T_{\bar{x}}\overline{\mathcal{W}} | \bar{g}_{\bar{x}}((\zeta_{\mathbf{U}}, \zeta_{\mathbf{Z}},), (\mathbf{U\Omega}, \mathbf{Z\Omega})) = 0\}. \tag{19}$$

Starting from an arbitrary tangent vector $\bar{\eta}_{\bar{x}} \in T_{\bar{x}}\overline{\mathcal{W}}$ we construct its projection on the horizontal space by picking $\mathbf{\Omega} \in S_{skew}(p)$ such that

$$\Pi_{\bar{x}}(\bar{\eta}_{\bar{x}}) = \quad (\bar{\eta}_{\mathbf{U}} - \mathbf{U\Omega}, \bar{\eta}_{\mathbf{Z}} - \mathbf{Z\Omega}) \quad \in \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}, \tag{20}$$

Using the calculation (19), the unique $\mathbf{\Omega}$ that satisfies (20) is the solution of the following set of *Lyapunov* equations. We introduce an auxiliary variable $\mathbf{\Omega}_{\text{dummy}} \in S_{skew}(r)$ that simplifies the exposition for finding $\mathbf{\Omega}$. The expression after introducing $\mathbf{\Omega}_{\text{dummy}}$ is

$$\begin{aligned}
&(\mathbf{Z}^T\mathbf{Z})\mathbf{\Omega}_{\text{dummy}} + \mathbf{\Omega}_{\text{dummy}}(\mathbf{Z}^T\mathbf{Z}) = 2\text{Skew}((\mathbf{Z}^T\mathbf{Z})(\mathbf{U}^T\bar{\eta}_{\mathbf{U}})(\mathbf{Z}^T\mathbf{Z})) - 2\text{Skew}((\bar{\eta}_{\mathbf{Z}}^T\mathbf{Z})(\mathbf{Z}^T\mathbf{Z})) \\
&(\mathbf{Z}^T\mathbf{Z})\mathbf{\Omega} + \mathbf{\Omega}(\mathbf{Z}^T\mathbf{Z}) = \mathbf{\Omega}_{\text{dummy}}
\end{aligned} \tag{21}$$

where the operator $\text{Skew}(\mathbf{A}) = \frac{\mathbf{A}-\mathbf{A}^T}{2}$. The numerical complexity of solving a Lyapunov equation is $O(r^3)$.

The Riemannian submersion $(\mathcal{W}, g)$

The choice of the metric (18), which is invariant along the equivalence class $[\bar{x}]$, and of the horizontal space (19) turn the quotient manifold $\mathcal{W}$ into a Riemannian submersion of $(\overline{\mathcal{W}}, \bar{g})$ [AMS08]. As shown in [AMS08], this special construction allows for a convenient matrix representation of the gradient and the Hessian on the abstract manifold $\mathcal{W}$. The Riemannian gradient of $\phi : \mathcal{W} \to \mathbb{R} : x \mapsto \phi(x) = \bar{\phi}(\bar{x})$ is uniquely represented by its horizontal lift in $\overline{\mathcal{W}}$ which has the matrix representation

$$\overline{\text{grad}_x\phi} = \text{grad}_{\bar{x}}\bar{\phi}. \tag{22}$$

It should be emphasized that $\text{grad}_{\bar{x}}\bar{\phi}$ is in the the tangent space $T_{\bar{x}}\overline{\mathcal{W}}$. However, due to invariance of the cost along the equivalence class $[\bar{x}]$, $\text{grad}_{\bar{x}}\bar{\phi}$ also belongs to the horizontal space $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ and hence, the equality in (22) [AMS08]. On the other hand, the matrix expression of $\text{grad}_{\bar{x}}\bar{\phi}$ at a point $\bar{x} = (\mathbf{U}, \mathbf{Z})$ is standard: the Euclidean gradient $(\text{Grad}_{\mathbf{U}}\bar{\phi}, \text{Grad}_{\mathbf{Z}}\bar{\phi})$ must simply be projected on $T_{\bar{x}}\overline{\mathcal{W}}$, i.e.,

$$\text{grad}_{\bar{x}}\bar{\phi} = \Psi_{\bar{x}}(\text{Grad}_{\mathbf{U}}\bar{\phi}, \text{Grad}_{\mathbf{Z}}\bar{\phi})$$

Likewise, the Riemannian connection $\nabla_\nu\eta$ on $\mathcal{W}$ is uniquely represented by its horizontal lift in $\overline{\mathcal{W}}$ which is

$$\overline{\nabla_\nu\eta} = \Pi_{\bar{x}}(\overline{\nabla}_{\bar{\nu}}\bar{\eta}).$$

where ν and η are vector fields in $\mathcal{W}$ and $\bar{\nu}$ and $\bar{\eta}$ are their horizontal lifts in $\overline{\mathcal{W}}$. In this case as well, the Riemannian connection $\overline{\nabla}_{\bar{\nu}}\bar{\eta}$ on $\overline{\mathcal{W}}$ has well-known expression [Jou09, Mey11, AMS08], obtained by means of the *Koszul formula*, given by

$$\overline{\nabla}_{\bar{\nu}}\bar{\eta} = \Psi_{\bar{x}}(\mathrm{D}\bar{\eta}[\bar{\nu}]) - \Psi_{\bar{x}} \begin{pmatrix} \bar{\nu}_{\mathbf{U}}\mathrm{Sym}(\mathbf{U}^T\bar{\eta}_{\mathbf{U}}), \\ -\ \bar{\eta}_{\mathbf{Z}}\big(\mathbf{Z}^T\mathbf{Z}\big)^{-1}\mathrm{Sym}\big(\mathbf{Z}^T\bar{\nu}_{\mathbf{Z}}\big) - \bar{\nu}_{\mathbf{Z}}\big(\mathbf{Z}^T\mathbf{Z}\big)^{-1}\mathrm{Sym}\big(\mathbf{Z}^T\bar{\eta}_{\mathbf{Z}}\big) \\ +\ \mathbf{Z}\big(\mathbf{Z}^T\mathbf{Z}\big)^{-1}\mathrm{Sym}\big(\bar{\eta}_{\mathbf{Z}}^T\bar{\nu}_{\mathbf{Z}}\big) \end{pmatrix} \tag{23}$$

Here $\mathrm{D}\bar{\eta}[\bar{\nu}]$ is the classical Euclidean directional derivative of $\bar{\eta}$ in the direction $\bar{\nu}$. The horizontal lift of the Riemannian Hessian in $\mathcal{W}$ has, thus, the following matrix expression

$$\overline{\mathrm{Hess}\phi(x)[\xi]} = \Pi_{\bar{x}}(\overline{\nabla}_{\bar{\xi}}\overline{\mathrm{grad}\phi}). \tag{24}$$

for any $\xi \in T_x\mathcal{W}$ and its horizontal lift $\bar{\xi} \in \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$.

Trust-region subproblem and retraction

The trust-region subproblem on the abstract space $\mathcal{W}$ is horizontally lifted to $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ and formulated as

$$\begin{array}{ll} \min_{\bar{\xi}\in\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}} & \bar{\phi}(\bar{x}) + g_{\bar{x}}(\bar{\xi}, \overline{\mathrm{grad}\phi(x)}) + \frac{1}{2}g_x(\bar{\xi}, \overline{\mathrm{Hess}\phi(x)[\xi]}) \\ \text{subject to} & g_{\bar{x}}(\bar{\xi},\bar{\xi}) \leq \delta. \end{array} \tag{25}$$

where δ is the trust-region radius and $\overline{\mathrm{grad}\phi}$ and $\overline{\mathrm{Hess}\phi}$ are the horizontal lifts of the Riemannian gradient and Hessian on $\mathcal{W}$. The trust-region subproblem is solved efficiently by the generic solver GenRTR [BAG07]. Refer [ABG07, BAG07] for the algorithmic details. The output is a direction $\bar{\xi}$ that minimizes the model.

To find the new iterate based on the obtained direction $\bar{\xi}$, a mapping from the tangent space $\mathcal{H}_{\bar{x}}$ to the manifold $\overline{\mathcal{W}}$ is required. This mapping is more generally referred to as *retraction* $R_x : \mathcal{H}_{\bar{x}}\overline{\mathcal{W}} \to \overline{\mathcal{W}}$ (details in [AMS08]). In the present case, a retraction of interest is [AMS08, Mey11]

$$\begin{aligned} R_{\mathbf{U}}(\bar{\xi}_{\mathbf{U}}) &= \mathrm{uf}(\mathbf{U} + \bar{\xi}_{\mathbf{U}}) \\ R_{\mathbf{Z}}(\bar{\xi}_{\mathbf{Z}}) &= \mathbf{Z} + \bar{\xi}_{\mathbf{Z}} \end{aligned} \tag{26}$$

where uf is a function that extracts the orthogonal factor [AMS08] i.e.,

$$\mathrm{uf}(\mathbf{A}) = \mathbf{A}(\mathbf{A}^T\mathbf{A})^{-\frac{1}{2}}.$$

Numerical complexity

The numerical complexity of manifold-based optimization depends on the computational cost of the following components.

- Objective function $\bar{\phi} \longrightarrow$ problem dependent
- Metric $\bar{g} \longrightarrow O(d_1r^2 + d_2r^2 + 2r^3)$
- Euclidean gradient of $\bar{\phi} \longrightarrow$ problem dependent

- $\overline{\nabla}_{\bar{\nu}}\bar{\eta}$
  - $\mathrm{D}\bar{\eta}[\bar{\nu}] \longrightarrow$ problem dependent
  - Matrix multiplication terms $\longrightarrow O(d_1r^2 + 3d_2r^2 + 5r^3)$
- Projection operator $\Psi \longrightarrow O(d_1r^2)$
- Projection operator $\Pi \longrightarrow O(d_1r^2 + d_2r^2)$
  - Solving Lyapunov equation $\longrightarrow O(r^3)$
- Retraction $R \longrightarrow O(d_1r^2 + 2r^3)$

As shown above all the manifold related operations are of linear complexity in d_1 and d_2. Other operations depend on the problem at hand and are computed in the product space $\overline{\mathcal{W}}$.

Discussion

The choice of (18) is motivated by the fact that the total space $\mathrm{St}(r, d_1) \times \mathbb{R}_*^{d_2 \times r}$ equipped with this metric is a complete Riemannian space. An alternative to (18) would be to consider the standard Euclidean metric

$$\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}}) \quad = \quad \mathrm{Tr}(\bar{\xi}_{\mathbf{U}}^T \bar{\eta}_{\mathbf{U}}) + \mathrm{Tr}(\bar{\xi}_{\mathbf{Z}}^T \bar{\eta}_{\mathbf{Z}}) \tag{27}$$

which is also invariant by rotations $\mathcal{O}(r)$. $\bar{\xi}_{\bar{x}}$ and $\bar{\eta}_{\bar{x}}$ are elements of $T_{\bar{x}}\overline{\mathcal{W}}$. This metric is for instance adopted in [SE10]. Although this alternative choice is appealing for its numerical simplicity, Figure 5 clearly illustrates the benefits of optimizing in a complete metric space. Under identical initializations and choice of step-size rule [NW06], the invariant metric prevents the numerical poor conditioning that arises in the presence of unbalanced factors $\mathbf{U}$ and $\mathbf{Z}$ i.e., $\|\mathbf{U}\|_F \not\approx \|\mathbf{Z}\|_F$.

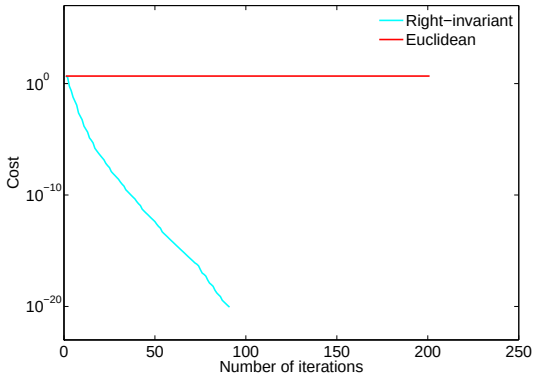

Figure 5: The choice of an invariant metric for subspace-projection factorization algorithm dramatically affects the algorithm performance. The example shown is to complete a rank 5 completion of a 4000×4000 matrix with 98% (OS = 8) entries missing but the observation is generic.

This factorization is also exploited in the recent paper [BA11] for the low-rank matrix completion problem (the application is described in Section 2.1). The RTRMC algorithm of [BA11] is an alternating minimization scheme where the authors exploit the fact that in the variable $\mathbf{Z}$, $\min_{\mathbf{Z}} \quad \bar{\phi}(\mathbf{U}, \mathbf{Z})$ is a least square problem that has a closed-form solution. They are, thus, left with an optimization problem in the other variable $\mathbf{U}$ on the Grassmann manifold $\mathrm{Gr}(r, d_1)$.

The resulting geometry of RTRMC is efficient in situations where $d_1 \ll d_2$ for the low-rank matrix completion. The advantage is reduced in square problems and the numerical experiments in Figure 6 suggest that our generic algorithm compares favorably to the Grassmanian algorithm in [BA11] in that case.

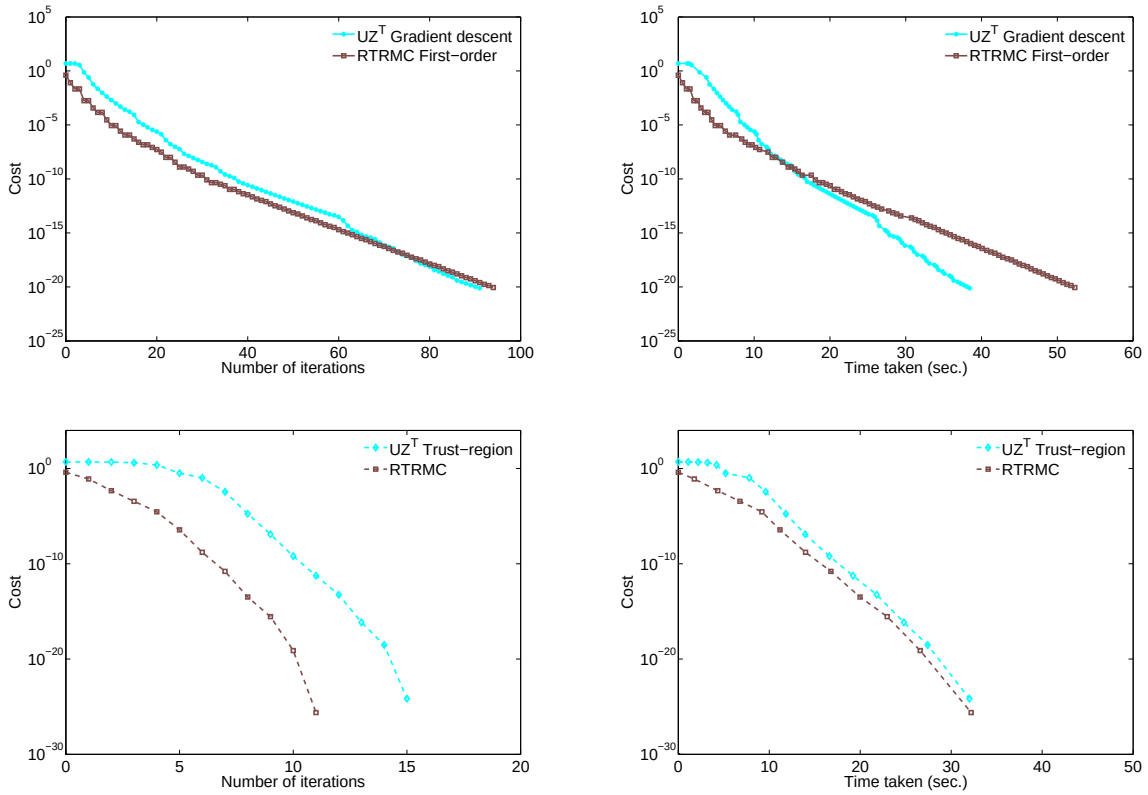

Figure 6: Rank 5 completion of 32000×32000 matrix with OS $= 8$. The framework proposed in this paper is competitive with RTRMC. *Top*: Comparison of gradient descent schemes. The quotient geometry uses the Armijo rule with adaptive step-size update for computing the step-size. RTRMC first-order uses the RTRMC code with Hessaian replaced by identity which yields a steepest descent algorithm [BA11]. *Below:* Comparison of trust-region schemes. Both algorithms use the solver GenRTR [BAG07] to solve the trust-regions subproblem.

5.2 Geometry of algorithms using full-rank factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$

The calculations in this part are similar to that of Section 5.1. Details are provided in [Mey11]. We provide here the final expressions for the ingredients that are needed for a a

generic optimization scheme.

The total space $\overline{\mathcal{W}}$ is $\mathbf{R}_*^{d_1\times r}\times\mathbf{R}_*^{d_2\times r}$ and due to the product structure the tangent space $T_{\bar{x}}\overline{\mathcal{W}}$ at $\bar{x}=(\mathbf{G},\mathbf{H})$ has the expression

$$T_{\bar{x}}\overline{\mathcal{W}}=\mathbf{R}^{d_1\times r}\times\mathbf{R}^{d_2\times r}.$$

Note that $\mathbb{R}_*^{d_1\times r}$ is an open subset of $\mathbb{R}^{d_1\times r}$ and thus, for an arbitrary matrix $(\mathbf{Z_G},\mathbf{Z_H})\in\mathbf{R}^{d_1\times r}\times\mathbf{R}^{d_2\times r}$, the projection operation $\Psi_{\bar{x}}$ onto the tangent space $T_{\bar{x}}\overline{\mathcal{W}}$ is the identity map

$$\Psi_{\bar{x}}(\mathbf{Z_G},\mathbf{Z_H})=(\mathbf{Z_G},\mathbf{Z_H}). \tag{28}$$

We endow the tangent space with the metric that is invariant to the equivalence classes generated by the invariance $\overline{\mathcal{W}}/\mathrm{GL}(r)$. We equip the subspaces, both *left* and *right*, with the right-invariant metric

$$\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}},\bar{\eta}_{\bar{x}}) \;=\; \mathrm{Tr}((\mathbf{G}^T\mathbf{G})^{-1}\bar{\xi}_{\mathbf{G}}^T\bar{\eta}_{\mathbf{G}})+\mathrm{Tr}((\mathbf{H}^T\mathbf{H})^{-1}\bar{\xi}_{\mathbf{H}}^T\bar{\eta}_{\mathbf{H}}) \tag{29}$$

where $\bar{\xi}_{\bar{x}},\bar{\eta}_{\bar{x}}\in T_{\bar{x}}\overline{\mathcal{W}}$. The tangent space is decomposed into the vertical and horizontal space with expressions

$$\begin{array}{l}\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}=\left\{\left(-\mathbf{G}\mathbf{\Lambda},\mathbf{H}\mathbf{\Lambda}^T\right)|\mathbf{\Lambda}\in\mathbb{R}^{r\times r}\right\}\quad\text{and}\\ \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}=\{\left(\bar{\zeta}_{\mathbf{G}},\bar{\zeta}_{\mathbf{H}}\right)|\bar{\zeta}_{\mathbf{G}}^T\mathbf{G}\mathbf{H}^T\mathbf{H}=\mathbf{G}^T\mathbf{G}\mathbf{H}^T\bar{\zeta}_{\mathbf{H}},\quad\bar{\zeta}_{\mathbf{G}}\in\mathbb{R}^{d_1\times r},\bar{\zeta}_{\mathbf{H}}\in\mathbb{R}^{d_2\times r}\}\end{array} \tag{30}$$

where $\bar{\zeta}_{\bar{x}}\in T_{\bar{x}}\overline{\mathcal{W}}$ and for any $\mathbf{\Lambda}\in\mathbb{R}^{r\times r}$. An element $\bar{\eta}_{\bar{x}}\in T_{\bar{x}}\overline{\mathcal{W}}$ is projected onto the horizontal space using the projection operator

$$\Pi_{\bar{x}}(\bar{\eta}_{\bar{x}})= \;\left(\bar{\eta}_{\mathbf{U}}+\mathbf{G}\mathbf{\Lambda},\bar{\eta}_{\mathbf{H}}-\mathbf{H}\mathbf{\Lambda}^T\right) \tag{31}$$

where $\mathbf{\Lambda}\in\mathbb{R}^{r\times r}$ is uniquely obtained by solving the following Lyapunov equation

$$\begin{array}{rll}\left(\bar{\eta}_{\mathbf{G}}+\mathbf{G}\mathbf{\Lambda}\right)^T\mathbf{G}\mathbf{H}^T\mathbf{H} & = & \mathbf{G}^T\mathbf{G}\mathbf{H}^T\left(\bar{\eta}_{\mathbf{H}}-\mathbf{H}\mathbf{\Lambda}^T\right)\\ \Rightarrow\quad\mathbf{\Lambda}^T\left(\mathbf{G}^T\mathbf{G}\mathbf{H}^T\mathbf{H}\right)+\left(\mathbf{G}^T\mathbf{G}\mathbf{H}^T\mathbf{H}\right)\mathbf{\Lambda}^T & = & \mathbf{G}^T\mathbf{G}\mathbf{H}^T\bar{\eta}_{\mathbf{H}}-\bar{\eta}_{\mathbf{G}}^T\mathbf{G}\mathbf{H}^T\mathbf{H}.\end{array} \tag{32}$$

Note that (32) is a Lyapunov equation of dimension r with the numerical cost $O(r^3)$. The expression of the Riemannian connection $\overline{\nabla}_{\bar{\nu}}\bar{\eta}$ in $\overline{\mathcal{W}}$ is

$$\begin{array}{rcl}\overline{\nabla}_{\bar{\nu}}\bar{\eta} & = & \mathrm{D}\bar{\eta}[\bar{\nu}]+(\mathbf{P},\mathbf{Q})\quad\text{and}\\ \mathbf{P} & = & -\bar{\eta}_{\mathbf{G}}\left(\mathbf{G}^T\mathbf{G}\right)^{-1}\mathrm{Sym}\left(\mathbf{G}^T\bar{\nu}_{\mathbf{G}}\right)-\bar{\nu}_{\mathbf{G}}\left(\mathbf{G}^T\mathbf{G}\right)^{-1}\mathrm{Sym}\left(\mathbf{G}^T\bar{\eta}_{\mathbf{G}}\right)\\ & & +\;\mathbf{G}\left(\mathbf{G}^T\mathbf{G}\right)^{-1}\mathrm{Sym}\left(\bar{\eta}_{\mathbf{G}}^T\bar{\nu}_{\mathbf{G}}\right),\\ \mathbf{Q} & = & -\bar{\eta}_{\mathbf{H}}\left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathrm{Sym}\left(\mathbf{H}^T\bar{\nu}_{\mathbf{H}}\right)-\bar{\nu}_{\mathbf{H}}\left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathrm{Sym}\left(\mathbf{H}^T\bar{\eta}_{\mathbf{H}}\right)\\ & & +\;\mathbf{H}\left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathrm{Sym}\left(\bar{\eta}_{\mathbf{H}}^T\bar{\nu}_{\mathbf{H}}\right).\end{array} \tag{33}$$

Here $\bar{\eta}$ and $\bar{\nu}$ are any vector fields on $\overline{\mathcal{W}}$ and $\mathrm{D}\bar{\eta}[\bar{\nu}]$ is the classical Euclidean directional derivative of $\bar{\eta}$ in the direction $\bar{\nu}$. Finally, we choose a retraction

$$\begin{array}{l}R_{\mathbf{G}}(\bar{\xi}_{\mathbf{G}})=\mathbf{G}+\bar{\xi}_{\mathbf{G}}\\ R_{\mathbf{H}}(\bar{\xi}_{\mathbf{H}})=\mathbf{H}+\bar{\xi}_{\mathbf{H}}\end{array} \tag{34}$$

As mentioned earlier in Section 3.1 we perform a *balancing update* on the retracted point to ensure better numerical conditioning. To balance the points $(\mathbf{G}, \mathbf{H})$ we use the following update rule

$$\begin{aligned} \mathbf{G}_{\text{balanced}} &= \frac{1}{\alpha}\mathbf{G}, \\ \mathbf{H}_{\text{balanced}} &= \alpha\mathbf{H}, \end{aligned} \tag{35}$$

where $\alpha \in \mathbb{R}_+$ is given by

$$\alpha = \sqrt{\|\mathbf{G}\|_F / \|\mathbf{H}\|_F}. \tag{36}$$

Observe that $\mathbf{G}_{\text{balanced}}\mathbf{H}_{\text{balanced}}^T = \mathbf{G}\mathbf{H}^T$. A different balancing rule is proposed in [MBS11a].

The resulting algorithm proceeds in two cascaded updates, see Figure 7. The first update moves the iterate from one equivalence class to another while minimizing the cost function of interest. The second "balancing" update is a change of representative along the given equivalence class. This scheme with cascaded retraction ensures that we asymptotically converge to a local minimum of the cost function with a balanced factorization.

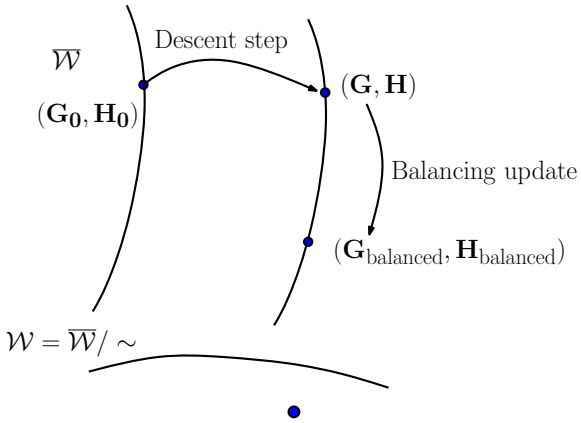

Figure 7: The proposed algorithm based on the factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ proceeds in two cascaded steps: a retraction step for cost minimization and a balancing step that ensures good numerical conditioning. The combination of these two steps correspond to a single iteration on the quotient manifold.

Discussion

The proposed scheme (gradient descent algorithm) is closely related to the gradient descent version of the Maximum Margin Matrix Factorization (MMMF) algorithm [RS05]. The gradient descent version of MMMF is a descent step in the product space $\mathbb{R}_*^{d_1 \times r} \times \mathbb{R}_*^{d_2 \times r}$ equipped with the Euclidean metric,

$$\bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}}) \quad = \quad \text{Tr}(\bar{\xi}_{\mathbf{G}}^T \bar{\eta}_{\mathbf{G}}) + \text{Tr}(\bar{\xi}_{\mathbf{H}}^T \bar{\eta}_{\mathbf{H}}) \tag{37}$$

where $\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}} \in T_{\bar{x}}\overline{\mathcal{W}}$. Note the difference with respect to (29). As a result, the invariance (with respect to $r \times r$ non-singular matrices) is not taken into account. In contrast, the proposed

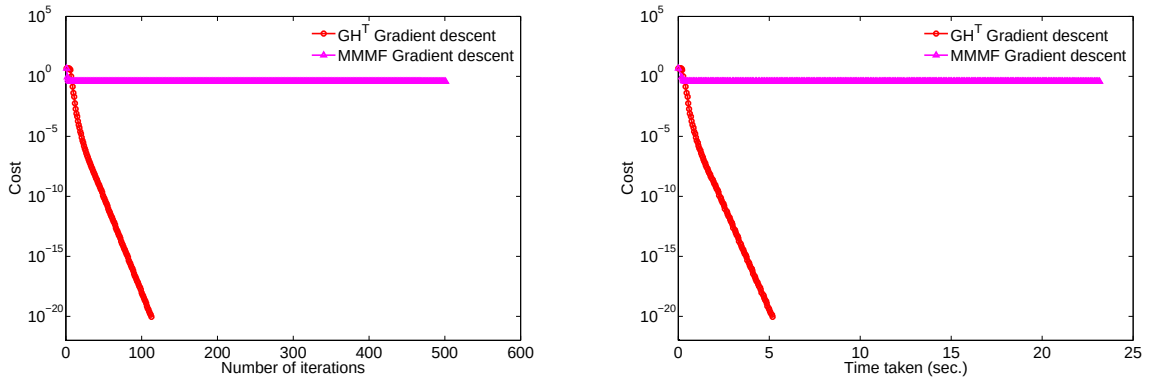

Figure 8: The proposed algorithm resolves the issue of choosing an appropriate step-size when there is a discrepancy between $\|\mathbf{G}\|_F$ and $\|\mathbf{H}\|_F$, a situation that leads to a slow convergence of the MMMF algorithm.

updates (34) and (35) are invariant along the set of equivalence classes (5). This resolves the issue of choosing an appropriate step size when there is a discrepancy between $\|\mathbf{G}\|_F$ and $\|\mathbf{H}\|_F$. Indeed, this situation leads to a slower convergence of the MMMF algorithm, whereas the proposed algorithm is not affected (Figure 8). To illustrate this effect, we consider a rank 5 matrix of size 4000×4000. An incomplete matrix is generated under standard assumptions. 2% of entries (OS = 8) are revealed uniformly at random. The step-size is computed using the Armijo rule [NW06]. Both algorithms are initialized similarly and the initial discrepancy between the factors at initialization is kept at $\|\mathbf{H}_0\|_F \approx 10^4 \|\mathbf{G}_0\|_F$.

The LMaFit algorithm [WYZ10] for the low-rank matrix completion problem relies on the factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ to alternatively learn the matrices $\mathbf{W}$, $\mathbf{G}$ and $\mathbf{H}$ so that the error $\|\mathbf{W} - \mathbf{G}\mathbf{H}^T\|_F^2$ is minimized while ensuring that the entries of $\mathbf{W}$ agree with the known entries i.e., $\mathcal{P}_\Omega(\mathbf{W}) = \mathcal{P}_\Omega(\mathbf{W}^\star)$. The algorithm is a tuned version of the block-coordinate descent algorithm in the product space that has a superior computational cost per iteration and better convergence than the straight forward non-linear coordinate scheme.

We compare both our gradient descent and trust-region algorithms with LMaFit in Figure 9. Though the numerical cost per iteration for the trust-region algorithm is higher than the gradient descent implementation, the trust-region algorithm is the algorithm of choice when a higher accuracy is required.

5.3 Geometry of algorithms using polar factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$

This section contains the final expressions of ingredients that are needed to perform both first-order and second-order optimization. Details are provided in the earlier references, [MMBS11, Mey11, BS09].

The total space $\overline{\mathcal{W}}$ is $\mathrm{St}(r, d_1) \times S_{++}(r) \times \mathrm{St}(r, d_2)$ and due to the product structure the tangent space $T_{\bar{x}}\overline{\mathcal{W}}$ at $\bar{x} = (\mathbf{U}, \mathbf{B}, \mathbf{V})$ is

$$T_{\bar{x}}\overline{\mathcal{W}} = T_{\mathbf{U}}\mathrm{St}(r, d_1) \times T_{\mathbf{B}}S_{++}(r) \times T_{\mathbf{V}}\mathrm{St}(r, d_2)$$

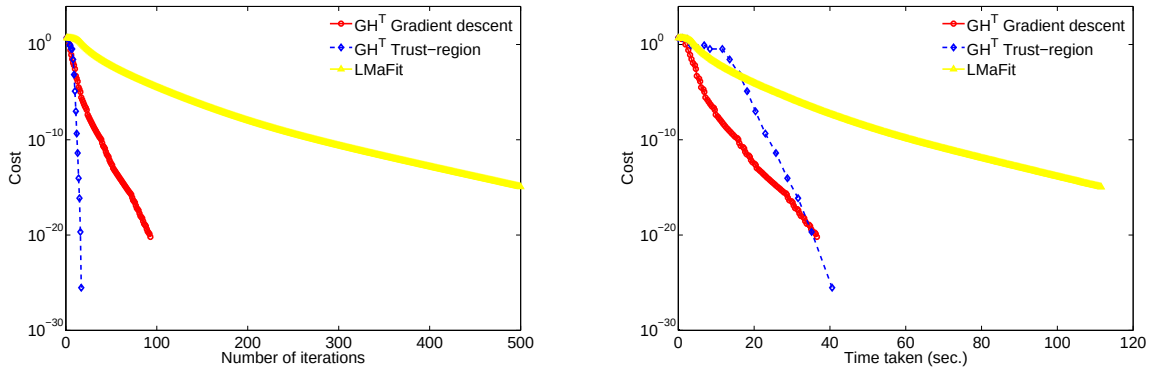

Figure 9: Rank 5 completion of 32000×32000 matrix with OS $= 8$. LMaFit has a superior computational complexity per iteration but the convergence seems to suffer for very large-scale matrices with low-rank.

and the following characterizations are well-known [EAS98, Smi05]

$$\begin{aligned} T_{\mathbf{U}}\mathrm{St}(r,d_1) &= \{\mathbf{U}\boldsymbol{\Omega} + \mathbf{U}_\perp\mathbf{K} | \boldsymbol{\Omega} \in S_{skew}(r), \mathbf{K} \in \mathbb{R}^{(d_1-r)\times r}\} \\ &= \{\mathbf{Z}_{\mathbf{U}} - \mathbf{U}\mathrm{Sym}(\mathbf{U}^T\mathbf{Z}_{\mathbf{U}}) | \mathbf{Z}_{\mathbf{U}} \in \mathbb{R}^{d_1\times r}\} \\ T_{\mathbf{B}}S_{++}(r) &= S_{sym}(r) \end{aligned}$$

where $S_{sym}(r)$ is the set of $r\times r$ symmetric matrices, $S_{skew}(r)$ is the set of $r\times r$ skew-symmetric matrices and $\mathbf{U}_\perp$ is the orthogonal complement of the space spanned by $\mathbf{U}$. Note that an arbitrary matrix $(\mathbf{Z}_{\mathbf{U}}, \mathbf{Z}_{\mathbf{B}}, \mathbf{Z}_{\mathbf{V}}) \in \mathbb{R}^{d_1\times r} \times \mathbb{R}^{r\times r} \times \mathbb{R}^{d_2\times r}$ is projected on the tangent space $T_{\bar{x}}\overline{\mathcal{W}}$ by the linear operation

$$\Psi_{\bar{x}}(\mathbf{Z}_{\mathbf{U}}, \mathbf{Z}_{\mathbf{B}}, \mathbf{Z}_{\mathbf{V}}) = (\mathbf{Z}_{\mathbf{U}} - \mathbf{U}\mathrm{Sym}(\mathbf{U}^T\mathbf{Z}_{\mathbf{U}}), \mathrm{Sym}(\mathbf{Z}_{\mathbf{B}}), \mathbf{Z}_{\mathbf{V}} - \mathbf{V}\mathrm{Sym}(\mathbf{V}^T\mathbf{Z}_{\mathbf{V}})). \tag{38}$$

The vertical space $\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}$ is the subspace of $T_{\bar{x}}\overline{\mathcal{W}}$ that is tangent to the equivalence class $[\bar{x}]$

$$\mathcal{V}_{\bar{x}}\overline{\mathcal{W}} = \{(\mathbf{U}\boldsymbol{\Omega}, \mathbf{B}\boldsymbol{\Omega} - \boldsymbol{\Omega}\mathbf{B}, \mathbf{V}\boldsymbol{\Omega}) | \boldsymbol{\Omega} \in S_{skew}(r)\}. \tag{39}$$

We choose $\mathcal{H}_{\bar{x}}\overline{\mathcal{W}}$ as the orthogonal complement of $\mathcal{V}_{\bar{x}}\overline{\mathcal{W}}$ for the metric

$$\begin{aligned} \bar{g}_{\bar{x}}(\bar{\xi}_{\bar{x}}, \bar{\eta}_{\bar{x}}) &= \mathrm{Tr}(\bar{\xi}_{\mathbf{U}}^T\bar{\eta}_{\mathbf{U}}) + \mathrm{Tr}(\mathbf{B}^{-1}\bar{\xi}_{\mathbf{B}}\mathbf{B}^{-1}\bar{\eta}_{\mathbf{B}}) \\ &\quad + \mathrm{Tr}(\bar{\xi}_{\mathbf{V}}^T\bar{\eta}_{\mathbf{V}}) \end{aligned} \tag{40}$$

where $\bar{\xi}_{\bar{x}}$ and $\bar{\eta}_{\bar{x}}$ are elements of $T_{\bar{x}}\mathcal{W}$. Starting from an arbitrary tangent vector $\bar{\eta}_{\bar{x}} \in T_{\bar{x}}\overline{\mathcal{W}}$ we construct its projection on the horizontal space by picking $\boldsymbol{\Omega} \in S_{skew}(r)$ such that

$$\Pi_{\bar{x}}(\bar{\eta}_{\bar{x}}) = \ (\bar{\eta}_{\mathbf{U}} - \mathbf{U}\boldsymbol{\Omega}, \bar{\eta}_{\mathbf{B}} - (\mathbf{B}\boldsymbol{\Omega} - \boldsymbol{\Omega}\mathbf{B}), \bar{\eta}_{\mathbf{V}} - \mathbf{V}\boldsymbol{\Omega}) \ \in \mathcal{H}_{\bar{x}}\overline{\mathcal{W}}, \tag{41}$$

The unique $\boldsymbol{\Omega}$ that satisfies (41) is the solution of the Lyapunov equation

$$\boldsymbol{\Omega}\mathbf{B}^2 + \mathbf{B}^2\boldsymbol{\Omega} = \mathbf{B}(\mathrm{Skew}(\mathbf{U}^T\bar{\eta}_{\mathbf{U}}) - 2\mathrm{Skew}(\mathbf{B}^{-1}\bar{\eta}_{\mathbf{B}}) + \mathrm{Skew}(\mathbf{V}^T\bar{\eta}_{\mathbf{V}}))\mathbf{B}. \tag{42}$$

The computational cost of solving the Lyapunov equation (42) is $O(r^3)$. Similar to the cases of other factorization models, the product structure of $\overline{\mathcal{W}}$ allows to write the Riemannian connection $\overline{\nabla}_{\bar{\nu}}\bar{\eta}$ in the total space using the expression [Jou09, Smi05, AMS08]

$$\overline{\nabla}_{\bar{\nu}}\bar{\eta} = \Psi_{\bar{x}}(\mathrm{D}\bar{\eta}[\bar{\nu}]) - \Psi_{\bar{x}}\left(\ \nu_{\mathbf{U}}\mathrm{Sym}(\mathbf{U}^T\eta_{\mathbf{U}}), \mathrm{Sym}(\nu_{\mathbf{B}}\mathbf{B}^{-1}\eta_{\mathbf{B}}), \nu_{\mathbf{V}}\mathrm{Sym}(\mathbf{V}^T\eta_{\mathbf{V}}) \ \right) \tag{43}$$

where $\mathrm{D}\bar{\eta}[\bar{\nu}]$ is the classical Euclidean directional derivative of $\bar{\eta}$ in the direction $\bar{\nu}$. Finally, a retraction of interest is [AMS08, MBS11a]

$$\begin{aligned} R_{\mathbf{U}}(\xi_{\mathbf{U}}) &= \mathrm{uf}(\mathbf{U} + \xi_{\mathbf{U}}) \\ R_{\mathbf{B}}(\xi_{\mathbf{B}}) &= \mathbf{B}^{\frac{1}{2}} \exp(\mathbf{B}^{-\frac{1}{2}} \xi_{\mathbf{B}} \mathbf{B}^{-\frac{1}{2}}) \mathbf{B}^{\frac{1}{2}} \\ R_{\mathbf{V}}(\xi_{\mathbf{V}}) &= \mathrm{uf}(\mathbf{V} + \xi_{\mathbf{V}}) \end{aligned} \tag{44}$$

where uf is a function that extracts the orthogonal factor (refer Section 5.1) and exp is the *matrix exponential* operator. The retraction on the positive definite cone $S_{++}(r)$ is the natural exponential mapping for the metric (11) [Smi05].

Discussion

We illustrate here the empirical evidence that constraining $\mathbf{B}$ to be diagonal (as is the case with singular value decomposition) is detrimental to optimization. We consider the simplest implementation of a gradient descent algorithm for matrix completion problem (see below). The plots shown in Figure 10 compare the behavior of the same algorithm in the search space $\mathrm{St}(r, d_1) \times S_{++}(r) \times \mathrm{St}(r, d_2)$ (polar factorization) and $\mathrm{St}(r, d_1) \times \mathrm{Diag}_+(r) \times \mathrm{St}(r, d_2)$ (SVD). $\mathrm{Diag}_+(r)$ is the set of diagonal matrices with positive entries. The metric and retraction updates are same for both the algorithms. The only difference lies in constraining $\mathbf{B}$ to be diagonal which means that the Riemannian gradient for the later case is in the space $\mathrm{Diag}(r)$ (the tangent space of $\mathrm{Diag}_+(r)$).

The empirical observation that convergence suffers from imposing diagonalization on $\mathbf{B}$ is a generic observation that does not depend on the particular problem at hand. The problem here involves completing a 4000×4000 of rank 5 from 2% of observed entries.

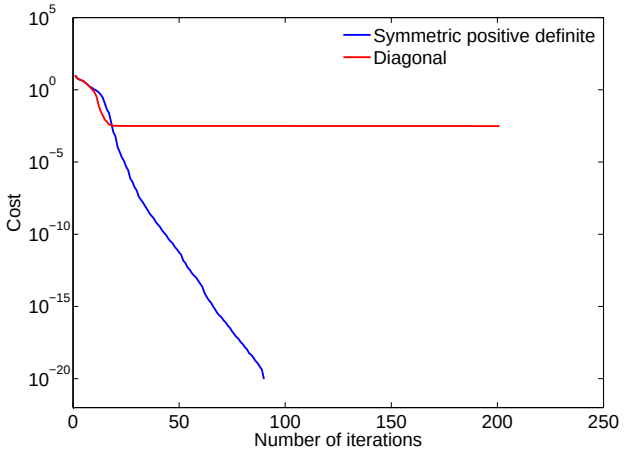

Figure 10: Convergence of a gradient descent algorithm is affected by making $\mathbf{B}$ diagonal.

The OptSpace algorithm [KMO10] also relies on the factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$, but with $\mathbf{B} \in \mathbb{R}^{r \times r}$. At each iteration, the algorithm minimizes the cost function $\bar{\phi}$

$$\min_{\mathbf{U},\mathbf{V}} \quad \bar{\phi}(\mathbf{U}, \mathbf{B}, \mathbf{V})$$

over the manifold $\mathrm{St}(r, d_1)/\mathcal{O}(r) \times \mathrm{St}(r, d_2)/\mathcal{O}(r)$ obtained by fixing $\mathbf{B}$ and then solving the inner optimization problem

$$\min_{\mathbf{B}} \quad \bar{\phi}(\mathbf{U}, \mathbf{B}, \mathbf{V}). \tag{45}$$

for fixed $\mathbf{U}$ and $\mathbf{V}$. The algorithm thus alternates between a gradient descent step on the subspaces $\mathbf{U}$ and $\mathbf{V}$ for fixed $\mathbf{B}$, and a least-square estimation of $\mathbf{B}$ (matrix completion problem) for fixed $\mathbf{U}$ and $\mathbf{V}$. The proposed framework is different from OptSpace in the choice $\mathbf{B}$ positive definite versus $\mathbf{B} \in \mathbb{R}^{r \times r}$. As a consequence, each step of the algorithm retains the geometry of polar factorization. Our algorithm also differs from OptSpace in the simultaneous and progressive nature of the updates. Furthermore, the choice $\mathbf{B} \succ 0$ allows us to derive alternative updates based on different metrics on the set $S_{++}(r)$. This flexibility is exploited in a previous paper [MBS11b] to show that metrics of $S_{++}(r)$ are connected to Bregman divergences and information geometry. A potential limitation of OptSpace comes from the fact that the inner optimization problem (45) may not be always solvable efficiently for other applications.

The singular value projection (SVP) algorithm [JMD10] is based on the SVD factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$ with $\mathbf{B} \in \mathrm{Diag}_+(r)$, diagonal matrix with positive entries. It can also be interpreted in the considered framework as a gradient descent algorithm in the embedding space $\mathbb{R}^{d_1 \times d_2}$, along with an efficient SVD-based retraction exploiting the sparse structure of the gradient $\xi_{\text{euclidean}}$ (gradient in the Euclidean space $\mathbb{R}^{d_1 \times d_2}$) for the matrix completion problem. A general update for SVP can be written as

$$\mathbf{U}_+\mathbf{B}_+\mathbf{V}_+^T = \mathrm{SVD}_r(\mathbf{U}\mathbf{B}\mathbf{V}^T - \xi_{\text{euclidean}}),$$

where $\mathrm{SVD}_r(\cdot)$ extracts r dominant singular values and singular vectors. An intrinsic limitation of the approach is that the computational cost of the algorithm is conditioned on the particular structure of the gradient. For instance, efficient routines exist for modifying the SVD with sparse [Lar98] or low-rank updates [Bra06].

Figure 11 shows the competitiveness of the proposed framework of polar factorization model with the SVP algorithm. The test example is an incomplete rank 5 matrix of size 32000×32000 with $\mathrm{OS} = 8$. We could not compare the performance of the OptSpace algorithm as some MATLAB operations (in the code supplied by the authors) have not been optimized for large matrices. We have, however, observed the good performance of the OptSpace algorithm on smaller problems.

5.4 Euclidean embeddings

Another view point of the set of fixed-rank matrices is the *embedded submanifold* view. The search space $\mathcal{F}(r, d_1, d_2)$ is the set of $r-\text{rank}$ $\mathbb{R}^{d_1 \times d_2}$ matrices. Recent papers [Van11, SWC10] investigate the search space in detail and develop both first-order and second-order algorithms. While conceptually the iterates move on the embedded space, numerically the implementation is done using factorization models, full-rank factorization [SWC10] and singular value decomposition [Van11]. We show the characterization of the tangent space using the factorization model $\mathbf{W} \in \mathcal{F}$ with singular value decomposition $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ where $\mathbf{\Sigma}$ is diagonal with

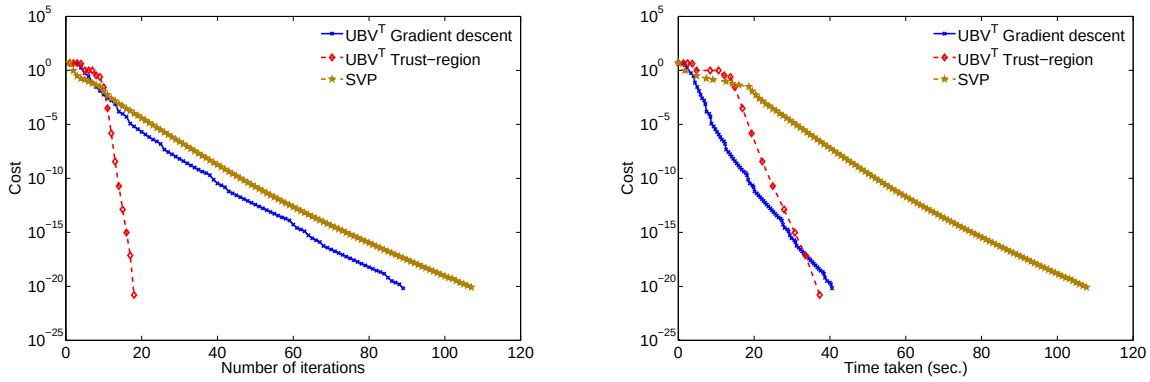

Figure 11: Illustration of the trust-region and gradient descent algorithms on low-rank matrix completion problem for polar factorization. The step-sizes for gradient descent algorithms are computed using the Armijo rule. For our gradient descent implementation we use (13) for an initial step-size guess. On the other hand, SVP uses an initial guess $\frac{d_1 d_2}{|\Omega|(1+\delta)}$ with $\delta = 1/3$ in [JMD10]. The main computational burden for SVP comes from computing the dominant singular value decomposition which is absent in the quotient geometry.

positive entries [Van11] and $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices. The treatment is similar for the factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ as the underlying geometry is the same [SWC10].

The tangent space expression at $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is given by

$$T_{\mathbf{W}}\mathcal{F} = \{\mathbf{U}\mathbf{M}\mathbf{V}^T + \mathbf{U}_p\mathbf{V}^T + \mathbf{U}\mathbf{V}_p^T | \mathbf{M} \in \mathbb{R}^{r\times r}, \mathbf{U}_p \in \mathbb{R}^{d_1\times r}, \mathbf{V}_p \in \mathbb{R}^{d_2\times r}\} \tag{46}$$

where $\mathbf{U}_p^T\mathbf{U} = \mathbf{0}$ and $\mathbf{V}_p^T\mathbf{V} = \mathbf{0}$. The set $\mathcal{F}$ is turned into a Riemannian submanifold of $\mathbb{R}^{d_1\times d_2}$ with the Riemannian metric

$$g_{\mathbf{W}}(\xi, \eta) = \mathrm{Tr}(\xi^T\eta) \tag{47}$$

which is the Euclidean inner product in $\mathbb{R}^{d_1\times d_2}$ now restricted to the set $\mathcal{F}$. ξ and η are tangent vectors. An arbitrary vector $\mathbf{Z} \in \mathbb{R}^{d_1\times d_2}$ is projected on the tangent space $T_{\mathbf{W}}\mathcal{F}$ is given using the operator $\Pi_{\mathbf{W}}$ by computing

$$\begin{aligned} \mathbf{M} &= \mathbf{U}\mathbf{Z}\mathbf{V} \\ \mathbf{U}_p &= \mathbf{Z}\mathbf{V} - \mathbf{U}\mathbf{M} \\ \mathbf{V}_p &= \mathbf{Z}^T\mathbf{U} - \mathbf{V}\mathbf{M}^T. \end{aligned} \tag{48}$$

A retraction is given by the formula

$$R_{\mathbf{W}}(\xi) = [\mathbf{U} \quad \mathbf{U}_p] \begin{bmatrix} \mathbf{\Sigma} + \mathbf{M} & \mathbf{I}_r \\ \mathbf{I}_r & \mathbf{0} \end{bmatrix} [\mathbf{V} \quad \mathbf{V}_p]^T. \tag{49}$$

Note that this can be computed efficiently due to singular value decomposition. The corresponding expressions for Riemannian gradient $\mathrm{grad}_{\mathbf{W}} f(\mathbf{W})$ for a smooth function $f : \mathcal{F} \to \mathbb{R} : \mathbf{W} \mapsto f(\mathbf{W})$ is given by

$$\mathrm{grad}_{\mathbf{W}} f = \Pi_{\mathbf{W}}(\mathrm{Grad}_{\mathbf{W}} f) \tag{50}$$

where $\text{Grad}_{\mathbf{W}} f$ is the gradient of f in the Euclidean space $\mathbb{R}^{d_1 \times d_2}$ at $\mathbf{W}$. Similarly, the expression of $\text{Hess} f(\mathbf{W})[\xi]$ in the direction $\xi \in T_{\mathbf{W}}\mathcal{F}$ at $\mathbf{W} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is also developed and specifically, for the matrix completion problem (2) is [Van11]

$$\begin{aligned} \text{Hess} f(\mathbf{X})[\xi] &= P_{\mathbf{U}}\mathcal{P}_\Omega(\xi)P_{\mathbf{V}} + P_{\mathbf{U}}^{\perp}(\mathcal{P}_\Omega(\xi) + \mathcal{P}_\Omega(\mathbf{W} - \mathbf{W}^*)\mathbf{V}_p\mathbf{\Sigma}^{-1}\mathbf{V}^T)P_{\mathbf{V}} \\ &\quad + P_{\mathbf{U}}(\mathcal{P}_\Omega(\xi) + \mathbf{U}\mathbf{\Sigma}^{-1}\mathbf{U}_p^T\mathcal{P}_\Omega(\mathbf{W} - \mathbf{W}^*))P_{\mathbf{V}}^{\perp} \end{aligned} \tag{51}$$

where $\mathbf{W}^*$ is the matrix with partially revealed entreis, $\xi = \mathbf{U}\mathbf{M}\mathbf{V}^T + \mathbf{U}_p\mathbf{V}^T + \mathbf{U}\mathbf{V}_p^T \in T_{\mathbf{W}}\mathcal{F}$ and the projection operators are $P_{\mathbf{U}} = \mathbf{U}\mathbf{U}^T$, $P_{\mathbf{U}}^{\perp} = \mathbf{I} - P_{\mathbf{U}}$, $P_{\mathbf{V}} = \mathbf{V}\mathbf{V}^T$ and $P_{\mathbf{V}}^{\perp} = \mathbf{I} - P_{\mathbf{V}}$.

Discussion

The visualization of the search space as an embedded submanifold of $\mathbb{R}^{d_1 \times d_2}$ has some key advantages. For example, the notions of geometric objects can be interpreted in a straight forward way. In the matrix completion case, this allows to compute the initial guess for the Armijo line-search efficiently [Van11].

On the other hand, the product space representation of the total space seems naturally related to the matrix factorization and provides additional flexibility in the choice of the metric. It is only the projection to the horizontal space that couples the separated choice of a metric on each component of the product space. From the optimization point of view this flexibility is also of interest. For instance it allows for different regularization of the matrix factors. The numerical comparisons in Figure 12 suggest that quotient geometries are competitive. The acronyms Loreta and LRGeom stand for the algorithms proposed in [SWC10] and [Van11] respectively. Loreta is a gradient descent algorithm based on the factorization $\mathbf{W} = \mathbf{G}\mathbf{H}^T$. LRGeom has both first-order and trust-region implementations [Van11].

6 Conclusion

We have addressed the problem of rank-constrained optimization (1) and presented both first-order and second-order schemes. The proposed framework is general and encompasses recent advances in optimization algorithms. In particular, we have studied various well-known algorithms for rank-constrained optimization and shown that most of these algorithms are based on three factorization models. Each factorization model offers its own advantages and is viewed as the product space of smooth and well-studied manifolds. This results in interesting invariance properties in the search space. We equip the product space with Riemannian geometry and develop necessary tools to perform optimization. Geometry related operations are linear in the number of rows (or columns) and exact expressions have been developed for all three factorization models. Additionally, the product structure enables us to use different metrics (subject to the constraint of respecting invariance) on the search space. We have shown one practical usefulness of choosing a proper metric in the context of subspace-projection factorization in Section 5.1. The usefulness of smooth search space has been discussed in the context of polar factorization in Section 5.3 where the flexibility of $\mathbf{B}$ to be positive definite results in good convergence properties. Similarly, the advantage of

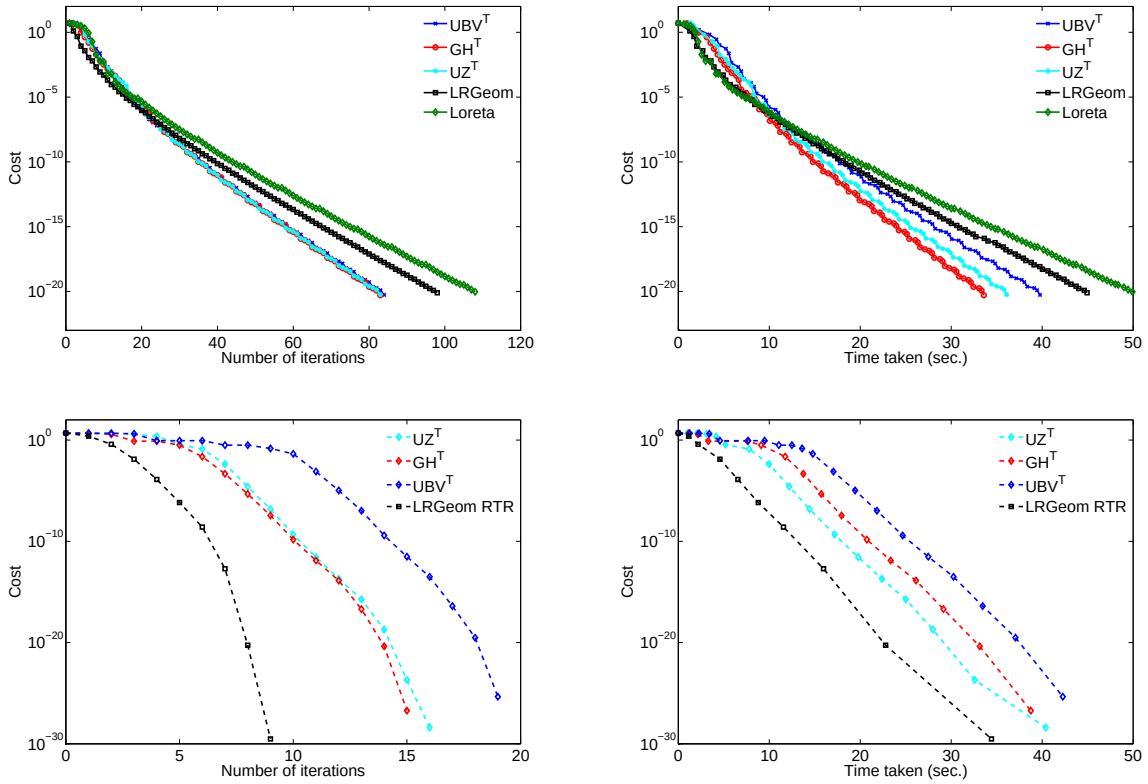

Figure 12: Low-rank matrix completion of size 32000×32000 of rank 5 with OS $= 8$. Both quotient and embedded geometries behave similarly. *Top:* Gradient descent algorithms. All algorithms are initialized similarly. The step-size for all algorithms is computed using Armijo rule along with adaptive step-size update (13). For LRGeom the step-size guess suggested in [Van11] is not used as it leads to a higher computational overload on the LRGeom algorithm than the adaptive step-size update. *Below:* Trust-region algorithms. The trust-region subproblem is solved using GenRTR [BAG07]. In many instances LRGeom has a better rate of convergence.

balancing an update has been discussed in Section 5.2. We have illustrated the usefulness of our framework on the application of low-rank matrix completion where we compete with the state-of-the-art algorithms on various benchmarks while maintaining a complete generality (with smoothness) of the cost function.

So, which matrix factorization to use? This depends primarily on the application. If computational complexity (per iteration) is a criterion, then algorithms based on $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ have a slight advantage over $\mathbf{W} = \mathbf{U}\mathbf{Z}^T$ and $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$, in this order, owing to a cheaper retraction and projection operators in the space of full-rank matrices $\mathbb{R}_*^{d\times r}$ than in $\mathrm{St}(r, d)$. Apart from this, the mentioned matrix factorizations also differ in one other aspect, namely, *invariance* in the space of $\mathbb{R}^{d_1\times d_2}$. Simply put, given a matrix $\mathbf{W} \in \mathcal{F}(r, d_1, d_2)$, what are the invariant group actions on $\mathbf{W}$ for each of the factorization geometry. The factorization $\mathbf{W} = \mathbf{U}\mathbf{B}\mathbf{V}^T$ captures the largest class of transformations that can be made invariant and $\mathbf{W} = \mathbf{G}\mathbf{H}^T$ captures the least and $\mathbf{W} = \mathbf{U}\mathbf{Z}^T$ places itself in between. Tuning the geometry

and the metric to particular problems will be a topic of future research.

References

[ABEV09] J. Abernethy, F. Bach, T. Evgeniou, and J.-P. Vert, *A new approach to collaborative filtering: Operator estimation with spectral regularization*, Journal of Machine Learning Research **10** (2009), no. Mar, 803–826.

[ABG07] P.-A. Absil, C. G. Baker, and K. A. Gallivan, *Trust-region methods on Riemannian manifolds*, Foundations of Computational Mathematics **7** (2007), no. 3, 303–330.

[AFSU07] Y. Amit, M. Fink, N. Srebro, and S. Ullman, *Uncovering shared structures in multiclass classification*, Proceedings of the 24th International Conference on Machine Learning, 2007.

[AILH09] P.-A. Absil, M. Ishteva, L. De Lathauwer, and S. Van Huffel, *A geometric Newton method for Oja's vector field*, Neural Computation **21** (2009), 1415–1433.

[AMS04] P.-A. Absil, R. Mahony, and R. Sepulchre, *Riemannian geometry of Grassmann manifolds with a view on algorithmic computation*, Acta Appl. Math. **80** (2004), no. 2, 199–220.

[AMS08] ———, *Optimization algorithms on matrix manifolds*, Princeton University Press, 2008.

[BA11] N. Boumal and P.-A. Absil, *RTRMC: A Riemannian trust-region method for low-rank matrix completion*, Neural Information Processing Systems conference, NIPS, 2011.

[BAG07] C. G. Baker, P.-A. Absil, and K. A. Gallivan, *GenRTR: the Generic Riemannian Trust-region package*, 2007.

[Bra06] M. Brand, *Fast low-rank modifications of the thin singular value decomposition*, Linear Algebra and its Applications **415** (2006), no. 1, 20 – 30.

[BS09] S. Bonnabel and R. Sepulchre, *Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank*, SIAM Journal on Matrix Analysis and Applications **31** (2009), no. 3, 1055–1070.

[BY09] K. Bleakley and Y. Yamanishi, *Supervised prediction of drug-target interactions using bipartite local models*, Bioinformatics **25** (2009), no. 18, 2397–2403.

[CCS08] J.-F. Cai, E. J. Candes, and Z. Shen, *A singular value thresholding algorithm for matrix completion*, SIAM Journal on Optimization **20** (2008), no. 4, 1956–1982.

[CHH07] D. Cai, X. He, and J. Han, *Efficient kernel discriminant analysis via spectral regression*, ICDM, 2007.

[CR08] E. J. Candès and B. Recht, *Exact matrix completion via convex optimization*, Foundations of Computational Mathematics **9** (2008), 717–772.

[EAS98] A. Edelman, T.A. Arias, and S.T. Smith, *The geometry of algorithms with orthogonality constraints*, SIAM Journal on Matrix Analysis and Applications **20** (1998), no. 2, 303–353.

[EMP05] T. Evgeniou, C.A. Micchelli, and M. Pontil, *Learning multiple tasks with kernel methods*, Journal of Machine Learning Research **6** (2005), no. Apr, 615–637.

[Gro11] D. Gross, *Recovering low-rank matrices from few coefficients in any basis*, IEEE Transaction on Information Theory **57** (2011), no. 3, 1548–1566.

[GVL96] G. H. Golub and C. F. Van Loan, *Matrix computations*, The Johns Hopkins University Press, 1996.

[JMD10] P. Jain, R. Meka, and I. Dhillon, *Guaranteed rank minimization via singular value projection*, Advances in Neural Information Processing Systems 23 (J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, eds.), 2010, pp. 937–945.

[Jou09] M. Journée, *Geometric algorithms for component analysis with a view to gene expression data analysis*, Ph.D. thesis, University of Liège, Liège, Belgium, 2009.

[KMO10] R. H. Keshavan, A. Montanari, and S. Oh, *Matrix completion from noisy entries*, Journal of Machine Learning Research **11** (2010), no. Jul, 2057–2078.

[KSD09] B. Kulis, M. Sustik, and I. S. Dhillon, *Low-rank kernel learning with Bregman matrix divergences*, Journal of Machine Learning Research **10** (2009), 341–376.

[KSD11] B. Kulis, K. Saenko, and T. Darrell, *What you saw is not what you get: Domain adaptation using asymmetric kernel transforms*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011.

[Lar98] R.M. Larsen, *Lanczos bidiagonalization with partial reorthogonalization*, Technical Report DAIMI PB-357, Department of Computer Science, Aarhus University, 1998.

[LB09] Kiryung Lee and Yoram Bresler, *Admira: Atomic decomposition for minimum rank approximation*, arXiv:0905.0044v2 (2009).

[MBS11a] G. Meyer, S. Bonnabel, and R. Sepulchre, *Linear regression under fixed-rank constraints: a Riemannian approach*, Proceedings of the 28th International Conference on Machine Learning (ICML), 2011.

[MBS11b] ———, *Regression on fixed-rank positive semidefinite matrices: a Riemannian approach*, Journal of Machine Learning Research **11** (2011), no. Feb, 593–625.

[Mey11] G. Meyer, *Geometric optimization algorithms for linear regression on fixed-rank matrices*, Ph.D. thesis, University of Liège, 2011.

[MHT10] R. Mazumder, T. Hastie, and R. Tibshirani, *Spectral regularization algorithms for learning large incomplete matrices*, Journal of Machine Learning Research **11** (2010), no. Aug, 2287–2322.

[MJD09] Raghu Meka, Prateek Jain, and Inderjit S Dhillon, *Matrix completion from power-law distributed samples*, Advances in Neural Information Processing Systems 22 (Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, eds.), 2009, pp. 1258–1266.

[MMBS11] B. Mishra, G. Meyer, F. Bach, and R. Sepulchre, *Low-rank optimization with trace norm penalty*, Tech. report, arXiv.com, 2011.

[MMS11] B. Mishra, G. Meyer, and R. Sepulchre, *Low-rank optimization for distance matrix completion*, Proceedings of the 50th IEEE Conference on Decision and Control, Orlando (USA), 2011.

[NW06] J. Nocedal and S. J. Wright, *Numerical optimization, second edition*, Springer, 2006.

[PO99] R. Piziak and P. L. Odell, *Full rank factorization of matrices*, Mathematics Magazine **72** (1999), no. 3, 193–201.

[RS05] J. Rennie and N. Srebro, *Fast maximum margin matrix factorization for collaborative prediction*, Proceedings of the 22nd International Conference on Machine learning, 2005, pp. 713–719.

[SE10] L. Simonsson and L. Eldén, *Grassmann algorithms for low rank approximation of matrices with missing values*, BIT Numerical Mathematics **50** (2010), no. 1, 173–191.

[Smi05] S.T. Smith, *Covariance, subspace, and intrinsic Cramér-Rao bounds*, IEEE Transactions on Signal Processing **53** (2005), 1610–1630.

[SWC10] U. Shalit, D. Weinshall, and G. Chechik, *Online learning in the manifold of low-rank matrices*, Advances in Neural Information Processing Systems 23 (J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, eds.), 2010, pp. 2128–2136.

[Van11] B. Vandereycken, *Low-rank matrix completion by Riemannian optimization*, Tech. report, École Polytechnique Fédérale de Lausanne, EPFL, 2011.

[WYZ10] Z. Wen, W. Yin, and Y. Zhang, *Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm*, Tech. report, Rice University, 2010.

[YAG+08] Y. Yamanishi, M. Araki, A. Gutteridge, W. Honda, and M. Kanehisa, *Prediction of drug-target interaction networks from the integration of chemical and genomic spaces*, Bioinformatics **24** (2008), no. 13, i232.

[YELM07] M. Yuan, A. Ekici, Z. Lu, and R.D.C. Monteiro, *Dimension reduction and coefficient estimation in multivariate linear regression*, Journal of the Royal Statistical Society **69** (2007).